Risk-sensitive Inverse Reinforcement Learning via Semi- and Non-Parametric Methods

Sumeet Singh[1], Jonathan Lacotte[2], Anirudha Majumdar[3], and Marco Pavone[1]

[1]Department of Aeronautics and Astronautics, Stanford University *
[2]Department of Electrical Engineering, Stanford University †
[3]Department of Mechanical and Aerospace Engineering, Princeton University ‡

Abstract

The literature on Inverse Reinforcement Learning (IRL) typically assumes that humans take actions in order to minimize the expected value of a cost function, i.e., that humans are *risk neutral*. Yet, in practice, humans are often far from being risk neutral. To fill this gap, the objective of this paper is to devise a framework for *risk-sensitive* IRL in order to explicitly account for a human's risk sensitivity. To this end, we propose a flexible class of models based on *coherent risk measures*, which allow us to capture an entire spectrum of risk preferences from risk-neutral to worst-case. We propose efficient non-parametric algorithms based on linear programming and semi-parametric algorithms based on maximum likelihood for inferring a human's underlying risk measure and cost function for a rich class of static and dynamic decision-making settings. The resulting approach is demonstrated on a simulated driving game with ten human participants. Our method is able to infer and mimic a wide range of qualitatively different driving styles from highly risk-averse to risk-neutral in a data-efficient manner. Moreover, comparisons of the Risk-Sensitive (RS) IRL approach with a risk-neutral model show that the RS-IRL framework more accurately captures observed participant behavior both qualitatively and quantitatively, especially in scenarios where catastrophic outcomes such as collisions can occur.

1 Introduction

Imagine a world where robots and humans coexist and work seamlessly together. In order to realize this vision, robots should, among other things, be able to (1) accurately predict the actions of humans in their environment, (2) quickly learn the preferences of human agents in their proximity and act accordingly, and (3) learn how to accomplish new tasks from human demonstrations. Inverse Reinforcement Learning (IRL) (Russell, 1998; Ng and Russell, 2000; Abbeel and Ng, 2005; Levine and Koltun, 2012; Ramachandran and Amir, 2007; Ziebart et al., 2008; Englert and Toussaint, 2015) is a powerful and flexible framework for tackling these challenges and has been previously used for a wide range of tasks, including modeling and mimicking human driver behavior (Abbeel

*{ssingh19, pavone}@stanford.edu
†lacotte@stanford.edu
‡ani.majumdar@princeton.edu

and Ng, 2004; Kuderer et al., 2015; Sadigh et al., 2016a), pedestrian trajectory prediction (Ziebart et al., 2009; Mombaur et al., 2010; Kretzschmar et al., 2016), and legged robot locomotion (Zucker et al., 2010; Kolter et al., 2007; Park and Levine, 2013). More recently, the popular technique of Max-Entropy (MaxEnt) IRL, an inspiration for some of the techniques leveraged in this work, has been adopted in a deep learning framework (Wulfmeier et al., 2015), and embedded within the guided policy optimization algorithm (Finn et al., 2016). The underlying assumption behind IRL is that humans act optimally with respect to an (unknown) cost function. The goal of IRL is then to infer this cost function from observed actions of the human. By learning the human's underlying preferences (in contrast to, e.g., directly learning a policy for a given task), IRL allows one to generalize one's predictions to novel scenarios and environments.

The prevalent modeling assumption made by existing IRL techniques is that humans take actions in order to minimize the *expected value* of a random cost. Such a model, referred to as the expected value (EV) model, implies that humans are *risk neutral* with respect to the random cost; yet, humans are often far from being risk neutral. A generalization of the EV model is represented by the expected utility (EU) theory in economics (von Neumann and Morgenstern, 1944), whereby one assumes that a human is an optimizer of the expected value of a disutility function of a random cost. Despite the historical prominence of EU theory in modeling human behavior, a large body of literature from the theory of human decision making strongly suggests that humans behave in a manner that is *inconsistent* with the EU model. At a high level, the EU model has two main limitations: (1) experimental evidence consistently confirms that this model is lacking in its ability to describe human behavior in risky scenarios (Allais, 1953; Ellsberg, 1961; Kahneman and Tversky, 1979), and (2) the EU model assumes that humans make no distinction between scenarios in which the probabilities of outcomes are known and ones in which they are unknown, which is often not the case. Consequently, a robot interacting with a human in a safety-critical setting (e.g., autonomous driving or navigation using shared autonomy), while leveraging such an inference model, could make incorrect assumptions about the human agent's behavior, potentially leading to catastrophic outcomes.

The known and unknown probability scenarios are referred to as *risky* and *ambiguous* respectively in the decision theory literature. An elegant illustration of the role of ambiguity is provided by the *Ellsberg paradox* (Ellsberg, 1961). Imagine an urn (Urn 1) containing 50 red and 50 black balls. Urn 2 also contains 100 red and black balls, but the relative composition of colors is unknown. Suppose that there is a payoff of \$10 if a red ball is drawn (and no payoff for black). In human experiments, subjects display an overwhelming preference towards having a ball drawn from Urn 1. However, now suppose the subject is told that a black ball has \$10 payoff (and no payoff for red). Humans *still* prefer to draw from Urn 1. This is a *paradox*, since choosing to draw from Urn 1 in the first case (payoff for red) indicates that the human assesses the proportion of red in Urn 1 to be higher than in Urn 2, while choosing Urn 1 in the second case (payoff for black) indicates that the human assesses a lower proportion of red in Urn 1 than in Urn 2. Indeed, there is no utility function for the two outcomes that can resolve such a contradictory assessment of underlying probabilities since it stems from a subjective distortion of outcome *probabilities* rather than *rewards*.

The limitations of EU theory in modeling human behavior has prompted substantial work on various alternative theories such as rank-dependent expected utility (Quiggin, 1982), expected uncertain utility (Gul and Pesendorfer, 2014), dual theory of choice (distortion risk measures) (Yaari, 1987), prospect theory (Kahneman and Tversky, 1979; Barberis, 2013), and many more (see (Majumdar and Pavone, 2017) for a recent review of the various axiomatic underpinnings of these risk

measures). Further, one way to interpret the Ellsberg paradox is that humans are not only risk averse, but are also *ambiguity averse* – an observation that has sparked an alternative set of literature in decision theory on "ambiguity-averse" modeling; see, e.g., the recent review (Gilboa and Marinacci, 2016). It is clear that the assumptions made by EU theory thus represent significant restrictions from a modeling perspective in an IRL context since a human expert is likely to be both risk and ambiguity averse, especially in safety critical applications such as driving where outcomes are inherently ambiguous and can possibly incur very high cost.

The key insight of this paper is to address these challenges by modeling humans as evaluating costs according to an (unknown) *risk measure*. A risk measure is a function that maps an uncertain cost to a real number (the expected value is thus a particular risk measure and corresponds to risk neutrality). In particular, we will consider the class of *coherent risk measures* (CRMs) (Artzner et al., 1999; Shapiro, 2009; Ruszczyński, 2010). CRMs were proposed within the operations research community and have played an influential role within the modern theory of risk in finance (Rockafellar and Uryasev, 2000; Acerbi and Tasche, 2002; Acerbi, 2002; Rockafellar, 2007). This theory has also recently been adopted for risk-sensitive (RS) Model Predictive Control and decision making (Chow and Pavone, 2014; Chow et al., 2015), and guiding autonomous robot exploration for maximizing information gain in time-varying environments (Axelrod et al., 2016).

Coherent risk measures enjoy a number of useful properties that jointly provide key advantages over EV and EU theories in the context of IRL. First, they capture an entire spectrum of risk assessments from risk-neutral to worst-case and thus offer a significant degree of modeling flexibility. Second, they capture risk sensitivity in an *axiomatically justified* manner; specifically, they formally capture a number of intuitive properties that one would expect any risk measure should satisfy (see Section 2.2). Third, a representation theorem for CRMs (Section 2.2) implies that they can be interpreted as computing the expected value of a cost function in a worst-case sense over a *set* of probability distributions (referred to as the *risk envelope*). Thus, CRMs capture both risk and ambiguity aversion within the *same modeling framework* since the risk envelope can be interpreted as capturing uncertainty about the underlying probability distribution that generates outcomes in the world. Finally, they are tractable from a computational perspective; the representation theorem allows us to solve both the inverse and forward problems in a computationally tractable manner for a rich class of static and dynamic decision-making settings.

Statement of contributions: This paper presents an IRL algorithm that explicitly takes into account risk sensitivity under *general* axiomatically-justified risk models that jointly capture risk and ambiguity within the same modeling framework. To this end, this paper makes four primary contributions. First, we propose a flexible modeling framework for capturing risk sensitivity in humans by assuming that the human demonstrator (hereby referred to as the "expert") acts according to a CRM. This framework allows us to capture an entire spectrum of risk assessments from risk-neutral to worst-case. Second, we develop efficient algorithms based on Linear Programming (LP) for inferring an expert's underlying risk measure for a broad range of static (Section 3) decision-making settings, including a proof of convergence of the predictive capability of the algorithm in the case where we only attempt to learn the risk measure. We additionally consider cases where both the cost and risk measure of the expert are unknown. Third, we develop a maximum likelihood based model for inferring the expert's risk measure and cost function for a rich class of dynamic decision-making settings (Section 4), generalizing our work in (Majumdar et al., 2017). Fourth, we demonstrate our approach on a simulated driving game (visualized in Figure 1) using a state-of-the-art commercial driving simulator and present results on ten human participants (Section 5).

We show that our approach is able to infer and mimic qualitatively different driving styles ranging from highly risk-averse to risk-neutral using only a minute of training data from each participant. We also compare the predictions made by our risk-sensitive IRL (RS-IRL) approach with one that models the expert using expected value theory and demonstrate that the RS-IRL framework more accurately captures observed participant behavior both qualitatively and quantitatively, especially in scenarios involving significant risk to the participant-driven car.

(a) Visualization of simulator during the interactive game experiment as seen by participant.

(b) Logitech G29 game input hardware consists of a force-feedback steering wheel and accelerator and brake pedals.

Figure 1: The simulated driving game considered in this paper. The human controls the follower car using a force-feedback steering wheel and two pedals and must follow the leader (an "erratic driver") as closely as possible without colliding. We observed a wide range of behaviors from participants reflecting varying attitudes towards risk.

Related Work: Safety-critical control and decision making applications demand increased resilience to events of low probability and detrimental consequences (e.g., a UAV crashing due to unexpectedly large wind gusts or an autonomous car failing to accommodate for an erratic neighboring vehicle). Such problems have inspired the recent advancement of various restricted versions of the problems considered here. In particular, there is a large body of work on RS decision making. For instance, in (Howard and Matheson, 1972) the authors leverage the exponential (or entropic) risk. This has historically been a very popular technique for parameterizing risk-attitudes in decision theory but suffers from the usual drawbacks of the EU framework such as the calibration theorem (Rabin, 2000). The latter states that very little risk aversion over moderate costs leads to unrealistically high degrees of risk aversion over large costs, which is undesirable from a modeling perspective. Other RS Markov Decision Process (MDP) formulations include Markowitz-inspired mean-variance (Filar et al., 1989; Tamar et al., 2012), percentile criteria on objectives (Wu and Yuanlie, 1999) and constraints (Geibel and Wysotzki, 2005), and cumulative prospect theory (Prashanth et al., 2016). This has driven research in the design of learning-based solution algorithms, i.e., RS reinforcement learning (Mihatsch and Neuneier, 2002; Bäuerle and Ott, 2011; Tamar et al., 2012; Petrik and Subramanian, 2012; Shen et al., 2014; Tamar et al., 2016). Ambiguity in MDPs is also well studied via the robust MDP framework, see, e.g., (Nilim and El Ghaoui,

2005; Xu and Mannor, 2010), as well as (Osogami, 2012; Chow et al., 2015) where the risk and ambiguity duality of CRMs is exploited. The key difference between this literature and the present work is that we consider the *inverse* reinforcement learning problem.

Results in the RS-IRL setting are more limited and have largely been pursued in the *neuroeconomics* literature (Glimcher and Fehr, 2014). For example, (Hsu et al., 2005) performed Functional Magnetic Resonance Imaging (FMRI) studies of humans making decisions in risky and ambiguous settings and modeled risk and ambiguity aversion using parametric utility and weighted probability models. In a similar vein, (Shen et al., 2014) models risk aversion using utility based shortfalls (with utility functions fixed a priori) and presents FMRI studies on humans performing a sequential investment task. While this literature may be interpreted in the context of IRL, the models used to predict risk and ambiguity aversion are quite limited. Risk in (Sadigh et al., 2016b) is captured via a *single* parameter to represent the aggressiveness of the expert driver – a fairly limited model that additionally does not account for probabilistic uncertainty. More recently, the authors in (Ratliff and Mazumdar, 2017) leverage the shortfall-risk model and associated $Q-$value decomposition introduced in (Shen et al., 2014) to devise a gradient-based RS-IRL algorithm. The model again assumes an a priori known risk measure and parameterized utility function and the learning loss function is taken to be the likelihood of the observed actions assuming the Boltzmann distribution fit to the optimal $Q-$values. There are two key limitations of this approach. First, learning is performed assuming a known utility function and risk measure – both of which, in general, are difficult to fix *a priori* for a given application. Second, computing gradients involves taking expectations with respect to the optimal policy as determined by the current value of the parameters. This must be determined by solving the fixed-point equations defining the "forward" RL problem – a computationally demanding task for large or infinite domains. This limitation is not an artifact of RS-IRL but in fact a standard complexity issue in any MaxEnt IRL-based algorithm. In contrast, this work (1) harnesses the elegant dual representation results for CRMs to avoid having to assume a known risk measure, and (2) solves a significantly less complex forward problem by leveraging a receding-horizon planning model for the expert – a technique used to great effect also in (Sadigh et al., 2016a).

A first version of this work was presented in (Majumdar et al., 2017). In this revised and extended edition, we include the following additional contributions: (1) a significant improvement in the multi-step RS-IRL model which now accounts for an expert planning over sequential disturbance modes (as opposed to the single-stage model in (Majumdar et al., 2017)); (2) a formal proof of convergence guaranteeing that in the limit, the single-step RS-IRL model will exactly replicate the expert's behavior; (3) introduction of a new maximum likelihood based approach for inferring both the risk measure and cost function for the multi-step model without assuming any a priori functional form; (4) extensive experimental validation on a realistic driving simulator where we demonstrate a significant improvement in predictive performance enabled by the RS-IRL algorithm over the standard risk-neutral model.

2 Problem Formulation

2.1 Dynamics

Consider the following discrete-time dynamical system:

$$x_{k+1} = f(x_k, u_k, w_k), \tag{1}$$

where k is the time index, $x_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^m$ is the control input, and $w_k \in \mathcal{W}$ is the disturbance. The control input is assumed to be bounded component-wise: $u_k \in \mathcal{U} := \{u : u^- \leq u \leq u^+\}$. We take $\mathcal{W}$ to be a finite set $\{w^{[1]}, \ldots, w^{[L]}\}$ with probability mass function (pmf) $p := [p(1), p(2), \ldots, p(L)]$, where $\sum_{i=1}^{L} p(i) = 1$ and $p(i) > 0, \forall i \in \{1, \ldots, L\}$. The time-sampling of the disturbance w_k will be discussed in Section 4. We assume that we are given demonstrations from an *expert* in the form of sequences of state-control pairs $\{(x_k^*, u_k^*)\}_k$ and that the expert has knowledge of the underlying dynamics (1) and disturbance realizations $\mathcal{W}$, but not the disturbance pmf p. We will refer back to this assumption within the context of the experimental setting in Section 5.

2.2 Model of the Expert

We model the expert as a *risk-sensitive* decision-making agent acting according to a *coherent risk measure* (defined formally below). We refer to such a model as a *coherent risk model*.

We assume that the expert has a cost function $C(x_k, u_k)$ that captures his/her preferences about outcomes. Let Z denote the cumulative cost accrued by the agent when planning over some finite horizon into the future. Since the process $\{x_k\}$ is stochastic, Z is a random variable adapted to the sequence $\{x_k\}$. A *risk measure* is a function $\rho(Z)$ that maps this uncertain cost to a real number. We will assume that the expert is assessing risks according to a *coherent risk measure*, defined as,

Definition 1 (Coherent Risk Measures)**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $\mathcal{Z}$ be the space of random variables on Ω. A coherent risk measure (CRM) is a mapping $\rho : \mathcal{Z} \to \mathbb{R}$ that obeys the following four axioms. For all $Z, Z' \in \mathcal{Z}$:*

A1. Monotonicity: $Z \leq Z' \Rightarrow \rho(Z) \leq \rho(Z')$.

A2. Translation invariance: $\forall a \in \mathbb{R}$, $\rho(Z + a) = \rho(Z) + a$.

A3. Positive homogeneity: $\forall \lambda \geq 0$, $\rho(\lambda Z) = \lambda \rho(Z)$.

A4. Subadditivity: $\rho(Z + Z') \leq \rho(Z) + \rho(Z')$.

These axioms were originally proposed in (Artzner et al., 1999) to ensure the "rationality" of risk assessments. For example, A1 states that if a random cost Z is less than or equal to a random cost Z' *regardless of the disturbance realizations*, then Z must be considered less risky (one may think of the cost distributions Z and Z' stemming from different control policies). A4 reflects the intuition that a risk-averse agent should prefer to *diversify*. We refer the reader to (Artzner et al., 1999; Majumdar and Pavone, 2017) for a thorough justification of these axioms. We provide, below, a hallmark example of coherent risk measures, the Conditional Value-at-Risk (CVaR) at level $\alpha \in (0, 1]$.

For an integrable cost random variable $Z \in \mathcal{Z}$, let the quantity $v_{1-\alpha}(Z) := \inf\{z \in \mathbb{R} \,|\, \mathbb{P}(Z \leq z) \geq 1-\alpha\}$ denote its $(1-\alpha)$-quantile (also referred to as the Value-at-Risk, or VaR). For continuous distributions[1], $\text{CVaR}_\alpha(Z)$ is defined as:

$$\text{CVaR}_\alpha(Z) := \mathbb{E}[Z \,|\, Z \geq v_{1-\alpha}(Z)].$$

That is, $\text{CVaR}_\alpha(Z)$ is the expected value of the α-tail distribution of Z (see Figure 2). In particular, one can show that when $\alpha = 1$, $\text{CVaR}_\alpha(Z)$ reduces to the standard expected value $\mathbb{E}[Z]$. Thus, the expected value is a *special case* of CVaR. See (Shapiro et al., 2014, Chapter 6) for additional examples within this rich class of risk measures, which include CVaR, mean absolute semi-deviation, spectral risk measures, optimized certainty equivalent, and the distributionally robust risk.

[1] More general definitions can be found in (Rockafellar and Uryasev, 2002).

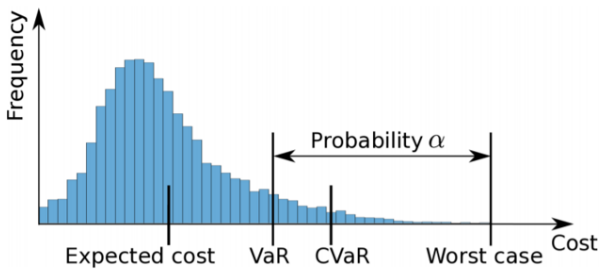


Figure 2: Illustration of the CVaR_α CRM. $\text{CVaR}_\alpha(Z)$ quantifies the mean of the α-tail of the cost distribution of Z.

An important characterization of CRMs is provided by the following representation theorem.

Theorem 1 (Representation Theorem for Coherent Risk Measures (Artzner et al., 1999))**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is a finite set with cardinality $|\Omega|$, $\mathcal{F}$ is the $\sigma-$algebra over subsets in Ω (i.e., $\mathcal{F} = 2^\Omega$), probabilities are assigned according to $\mathbb{P} = (p(1), p(2), \ldots, p(|\Omega|))$, and $\mathcal{Z}$ is the space of random variables on Ω. Denote by $\mathcal{C}$ the set of probability densities:*

$$\mathcal{C} := \left\{ \zeta \in \mathbb{R}^{|\Omega|} \mid \sum_{i=1}^{|\Omega|} p(i)\zeta(i) = 1,\ \zeta \geq 0 \right\}. \tag{2}$$

Define $q_\zeta \in \mathbb{R}^{|\Omega|}$ where $q_\zeta(i) = p(i)\zeta(i)$, $i = 1, \ldots, |\Omega|$. A risk measure $\rho : \mathcal{Z} \to \mathbb{R}$ with respect to the space $(\Omega, \mathcal{F}, \mathbb{P})$ is a CRM if and only if there exists a compact convex set $\mathcal{B} \subset \mathcal{C}$ such that for any $Z \in \mathcal{Z}$:

$$\rho(Z) = \max_{\zeta \in \mathcal{B}} \mathbb{E}_{q_\zeta}[Z] = \max_{\zeta \in \mathcal{B}} \sum_{i=1}^{|\Omega|} p(i)\zeta(i)Z(i). \tag{3}$$

This theorem is important for two reasons. Conceptually, it gives us an interpretation of CRMs as computing the worst-case expectation of the cost with respect to a *set* of *distorted* distributions $q_\zeta = p \cdot \zeta$. Coherent risk measures thus allow us to consider risk and ambiguity (see Section 1) in a unified framework since one may interpret an agent acting according to a coherent risk model as being *uncertain about the underlying probability density*. Practically, estimating this set of distributions provides us with an algorithmic handle for inferring the expert's risk preferences, and indeed will form the basis of our IRL methodology.

In this work, we will take the set $\mathcal{B}$ in (3) to be a polytope. We refer to such risk measures as *polytopic risk measures*, which were also considered in (Eichhorn and Römisch, 2005). Let $\Delta^{|\Omega|}$ denote the $|\Omega|-$dimensional probability simplex, defined as:

$$\Delta^{|\Omega|} := \{q \in \mathbb{R}^{|\Omega|} \mid \sum_{i=1}^{|\Omega|} q(i) = 1,\ q \geq 0\}.$$

By absorbing the density ζ into the pmf p in eq. (3), we can represent (without loss of generality) a polytopic risk measure as:

$$\rho(Z) = \max_{q \in \mathcal{P}} \mathbb{E}_q[Z], \tag{4}$$

where $\mathcal{P}$ is a polytopic subset of the probability simplex $\Delta^{|\Omega|}$:

$$\mathcal{P} = \left\{ q \in \Delta^{|\Omega|} \mid A_{\text{ineq}} q \leq b_{\text{ineq}} \right\}, \tag{5}$$

where the matrix $A_{\text{ineq}} \in \mathbb{R}^{d\times|\Omega|}$ and vector $b_{\text{ineq}} \in \mathbb{R}^d$ define a set of d halfspace constraints. The polytope $\mathcal{P}$ is hereby referred to as the *risk envelope*. Polytopic risk measures constitute a rich class of risk measures, encompassing a spectrum ranging from risk neutrality ($\mathcal{P} = \{p\}$) to worst-case assessments ($\mathcal{P} = \Delta^{|\Omega|}$); see also (Chow and Pavone, 2014; Shapiro et al., 2014). We further note that the *ambiguity* interpretation of CRMs is reminiscent of Gilboa & Schmeidler's Minmax EU model for ambiguity-aversion (Gilboa and Schmeidler, 1989) which was shown to outperform various competing models in (Hey et al., 2010) for single-stage decision problems, albeit with more restrictions on the set $\mathcal{B}$.

Goal: Given demonstrations from the expert in the form of state-control trajectories, the goal of this paper is to devise an algorithmic framework for risk-sensitive IRL whereby an expert's risk preferences will be estimated by finding an *approximation* of their risk envelope $\mathcal{P}$.

3 Risk-sensitive IRL: Single Decision Period

In this section we consider the single step decision problem. That is, given a current (known) state x_0, the expert chooses a single control action u_0 to minimize a coherent risk assessment of a random cost Z, represented by a non-negative cost function $C(x_1, u_0)$ where $x_1 = f(x_0, u_0, w_0)$. Thus, the uncertain cost Z is a random variable on the discrete probability space $(\mathcal{W}, 2^{\mathcal{W}}, p)$.

3.1 Known Cost Function

We first consider the static decision-making setting where the expert's cost function is known but the risk measure is unknown. A coherent risk model then implies that the expert is solving the following optimization problem at state x_0 in order to compute an optimal action:

$$\tau^* := \min_{u_0\in\mathcal{U}} \ \rho(C(x_1, u_0)) = \min_{u_0\in\mathcal{U}} \max_{q\in\mathcal{P}} \ \mathbb{E}_q[C(x_1, u_0)] \tag{6}$$

$$:= \min_{u_0\in\mathcal{U}} \max_{q\in\mathcal{P}} \ g(x_0, u_0)^T q, \tag{7}$$

where $g(x_0, u_0)(j)$ is the cost when the disturbance $w_0 = w^{[j]} \in \mathcal{W}$ is realized, and $\rho(\cdot)$ is a CRM with respect to the space $(\mathcal{W}, 2^{\mathcal{W}}, p)$ with risk envelope $\mathcal{P}$ being a subset of the probability simplex Δ^L. Since the inner maximization problem is linear in p, the optimal value is achieved at a vertex of the polytope $\mathcal{P}$. Let $\text{vert}(\mathcal{P}) = \{v_i\}$ denote the set of vertices of $\mathcal{P}$ and let N_V be the cardinality of this set. Then, we can rewrite problem (6) as:

$$\begin{aligned} \min_{u_0\in\mathcal{U},\tau} \quad & \tau \\ s.t. \quad & \tau \geq g(x_0, u_0)^T v_i, \quad i \in \{1, \dots, N_V\}. \end{aligned} \tag{8}$$

If the cost function $C(\cdot,\cdot)$ is convex in the control input u, the resulting optimization problem is convex. Given a dataset $\mathcal{D} = \{(x^{*,d}, u^{*,d})\}_{d=1}^D$ of state-control pairs of the expert taking action $u^{*,d}$ at state $x^{*,d}$, our goal is to deduce an approximation $\mathcal{P}_o$ of $\mathcal{P}$ from the given data. The key idea of our technical approach is to examine the Karush-Kuhn-Tucker (KKT) conditions for Problem (8). The use of KKT conditions for Inverse Optimal Control is a technique also adopted in (Englert and Toussaint, 2015). The KKT conditions are necessary for optimality in general and are also sufficient in the case of convex problems. We can thus use the KKT conditions along with the

dataset $\mathcal{D}$ to *constrain the constraints* of problem(8). In other words, the KKT conditions will allow us to constrain where the vertices of $\mathcal{P}$ must lie in order to be consistent with the fact that the state-control pairs represent optimal solutions to problem (8). Importantly, we will *not* assume access to the number of vertices N_V of $\mathcal{P}$.

Specifically, let (x^*, u^*) be an optimal state-control pair and let $\mathcal{J}^+$ and $\mathcal{J}^-$ denote the sets of components of the control input u^* that are saturated above and below respectively (i.e., $u^*(j) = u^+(j)$ for all $j \in \mathcal{J}^+$ and $u^*(j) = u^-(j)$ for all $j \in \mathcal{J}^-$).

Theorem 2 (KKT-Based Inference)**.** *Consider the following optimization problem:*

$$\begin{aligned} \max_{\substack{v\in\Delta^L \\ \sigma_+,\sigma_-\geq 0}} \quad & g(x^*,u^*)^T v \qquad (9)\\ s.t. \quad & 0 = \nabla_{u(j)} g(x,u)^T v\big|_{x^*,u^*} + \sigma_+(j), \forall j \in \mathcal{J}^+ \\ & 0 = \nabla_{u(j)} g(x,u)^T v\big|_{x^*,u^*} - \sigma_-(j), \forall j \in \mathcal{J}^- \\ & 0 = \nabla_{u(j)} g(x,u)^T v\big|_{x^*,u^*}, \forall j \notin \mathcal{J}^+, j \notin \mathcal{J}^- \\ & \sigma_+(j) = 0,\ \sigma_-(j) = 0, \quad \forall j \notin \mathcal{J}^+, j \notin \mathcal{J}^- \end{aligned}$$

Denote the optimal value of this problem by τ' and define the halfspace:

$$\mathcal{H}_{(x^*,u^*)} := \{v \in \mathbb{R}^L \mid \tau' \geq g(x^*,u^*)^T v\}. \qquad (10)$$

Then, the risk envelope $\mathcal{P}$ satisfies $\mathcal{P} \subset (\mathcal{H}_{(x^*,u^*)} \cap \Delta^L)$.

Proof. The KKT conditions for Problem (8) are:

$$1 = \sum_{i=1}^{N_V} \lambda_i, \qquad (11)$$

$$0 = \lambda_i [g(x^*,u^*)^T v_i - \tau],\ i = 1,\dots,N_V, \qquad (12)$$

and for $j = 1,\dots,m$:

$$0 = \sigma_+(j) - \sigma_-(j) + \sum_{i=1}^{N_V} \lambda_i\ \nabla_{u(j)} g(x,u)^T v_i\big|_{x^*,u^*}, \qquad (13)$$

$$0 = \sigma_+(j)[u^*(j) - u^+(j)],\ 0 = \sigma_-(j)[u^-(j) - u^*(j)], \qquad (14)$$

where $\lambda_i, \sigma_+(j), \sigma_-(j) \geq 0$ are multipliers. Now, suppose there are multiple optimal vertices $\{v_i\}_{i\in\mathcal{I}}$ for Problem (8) in the sense that $\tau^* = g(x^*,u^*)^T v_i$, for all $i \in \mathcal{I}$. Defining $\bar{v} := \sum_{i\in\mathcal{I}} \lambda_i v_i$, we see that $\bar{v}$ satisfies:

$$0 = \nabla_{u(j)} g(x^*,u^*(j))^T \bar{v} + \sigma_+(j) - \sigma_-(j), \quad j = 1,\dots,m, \qquad (15)$$

and $\tau^* = g(x^*,u^*)^T \bar{v}$ since $\sum_{i\in\mathcal{I}} \lambda_i = 1$. Now, since $\bar{v}$ satisfies the constraints of Problem (9) (which are implied by the KKT conditions), it follows that $\tau' \geq \tau^*$. From problem (8), we see that $\tau' \geq \tau^* \geq g(x^*,u^*)^T v_i$ for all $v_i \in \text{vert}(\mathcal{P})$ and thus $\mathcal{P} \subset (\mathcal{H}_{(x^*,u^*)} \cap \Delta^L)$. □

Problem (9) is a *Linear Program (LP)* and can thus be solved efficiently. For each demonstration $(x^{*,d}, u^{*,d}) \in \mathcal{D}$, Theorem 2 provides a halfspace constraint on the risk envelope $\mathcal{P}$. By aggregating these constraints, we obtain a *polytopic* outer approximation $\mathcal{P}_o$ of $\mathcal{P}$. This is summarized in Algorithm 1. Note that Algorithm 1 operates sequentially through the data $\mathcal{D}$ and is thus directly applicable in *online* settings. An illustration of the sequential pruning process in Algorithm 1 is provided in Figure 3.

Algorithm 1 Sequential Halfspace Pruning

1: Initialize $\mathcal{P}_o = \Delta^L$
2: **for** $d = 1, \ldots, D$ **do**
3: Solve Linear Program (9) with $(x^{*,d}, u^{*,d})$ to obtain a hyperplane $\mathcal{H}_{(x^{*,d}, u^{*,d})}$
4: Update $\mathcal{P}_o \leftarrow \mathcal{P}_o \cap \mathcal{H}_{(x^{*,d}, u^{*,d})}$
5: **end for**
6: Return $\mathcal{P}_o$

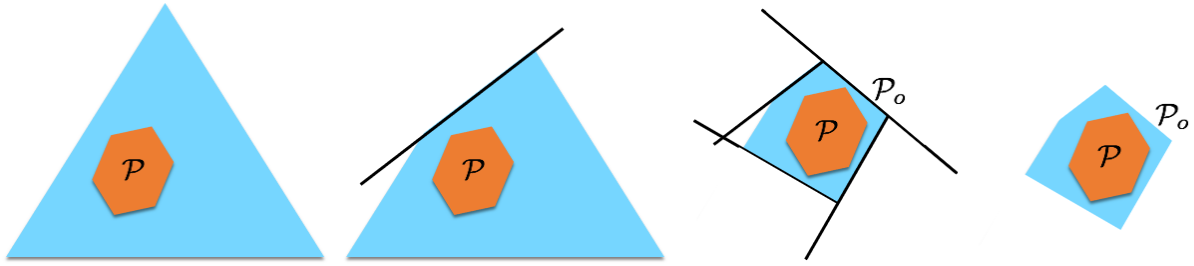


Figure 3: Schematic illustration of Algorithm 1. Probability simplex (3 scenarios) is shown in blue while the true (unknown) risk envelope is shown in orange. Algorithm 1 sequentially prunes portions of probability simplex that are inconsistent with the observed actions by leveraging the *necessary* conditions for optimality (KKT conditions) for problem (6). The end result is an outer approximation $\mathcal{P}_o$ of the true risk envelope $\mathcal{P}$.

Remark 1. *Algorithm 1 is a* non-parametric *algorithm for inferring the expert's risk measure; i.e., we are not fitting parameters for an a priori chosen risk measure. Instead, by leveraging the dual representation of CRMs as provided by Theorem 1 and reasoning directly over the risk envelope $\mathcal{P}$, Algorithm 1 can recover any risk measure within the class of CRMs, that best explains the expert's demonstrations.*

Remark 2. *As we collect more half-space constraints in Algorithm 1, the constraint $v \in \Delta^L$ in Problem (9) above can be replaced by $v \in \mathcal{P}_o$, where $\mathcal{P}_o$ is the current outer approximation of the risk envelope. It is easily verified that the results of Theorem 2 still hold. This allows us to obtain a tighter (i.e., lower) upper bound τ' for τ^*, thus resulting in tighter halfspace constraints for each new demonstration processed by the algorithm.*

Denote by $\mathcal{P}_D$ the output of Algorithm 1 after processing sequentially the first D demonstrations $\{(x^{*,d}, u^{*,d})\}_{d=1}^{D}$. Observe that for all $D \geq 1$, $\mathcal{P}_{D+1} \subseteq \mathcal{P}_D$. We can then define the limiting set as $\mathcal{P}_\infty := \bigcap_{d=1}^{\infty} \mathcal{P}_d$. An important consideration for this algorithm is whether it is possible to recover, at least from an imitation perspective, the risk envelope $\mathcal{P}$ from sufficiently many optimal demonstrations. In other words, we are specifically interested in the question of whether the limiting set $\mathcal{P}_\infty$ (whenever such a limit exists) allows one to *exactly* predict the actions of a decision maker

that operates under a risk model characterized by the set $\mathcal{P}$. In the following theorem we establish, under some restrictive technical conditions, that this is indeed possible. The proof is provided in Appendix A.

Theorem 3 (Convergence of Algorithm 1)**.** *Let $\mathcal{S} \subseteq \mathbb{R}^n$ be a convex, compact subset of the state space. Let $\{(x^{*,d}, u^{*,d})\}_{d=1}^{\infty}$ be a set of infinitely many optimal demonstrations such that the sequence $\{x^{*,d}\}$ is dense in $\mathcal{S}$. Assume that the following technical conditions hold:*

A.1 *The expert's cost vector $g(x, u)$ is strictly convex with respect to the control input u and continuous with respect to the state variable x.*

A.2 *For all $j \in \{1, \ldots, L\}$ and any state $x \in \mathcal{S}$, the cost function associated with the j-th disturbance $u \mapsto g(x, u)(j)$ has bounded level sets.*

Finally, for any risk envelope $\mathcal{P}' \subseteq \Delta^L$ and any state $x \in \mathcal{S}$, define

$$u(\mathcal{P}', x) := \operatorname*{argmin}_{u \in \mathcal{U}} \max_{v \in \mathcal{P}'} v^T g(x, u),$$

as the (unique) optimal control action of an expert with risk envelope $\mathcal{P}'$ at state x. Then, for any state $x \in \mathcal{S}$,

$$u(\mathcal{P}_\infty, x) = u(\mathcal{P}, x). \tag{16}$$

That is, for any state $x \in \mathcal{S}$, the optimal action predicted using the limiting envelope $\mathcal{P}_\infty$ matches that computed using the true expert polytope $\mathcal{P}$.

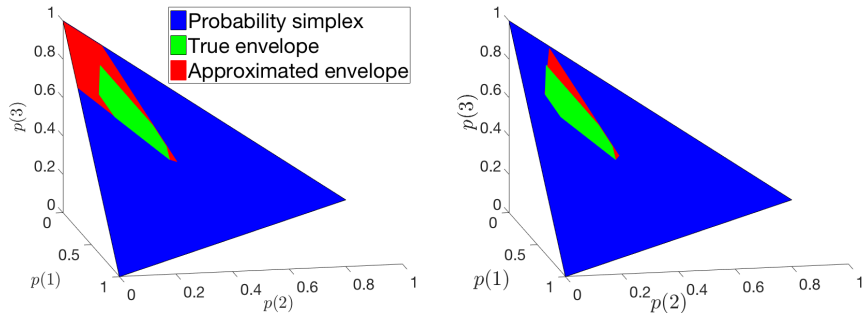
Remark 3. *The technical condition **A.1** assumes convexity of the cost vector with respect to the control input, and the proof of Theorem 3 heavily relies on this assumption. Finding conditions under which Algorithm 1 is guaranteed to be consistent for the general case of non-convex cost functions is an open problem left for future research; we re-emphasize, though, that Algorithm 1 is guaranteed to provide a conservative outer approximation regardless of the convexity of the cost vectors.*

Once we have recovered an approximation $\mathcal{P}_o$ of $\mathcal{P}$, we can solve the "forward" problem (i.e., compute actions at a given state x) by solving the optimization problem (6) with $\mathcal{P}_o$ as the risk envelope.

3.1.1 Example: Linear-Quadratic System

As a simple illustrative example to gain intuition for the convergence properties of Algorithm 1, consider a linear dynamical system with multiplicative uncertainty of the form $f(x_k, u_k, w_k) = A(w_k)x_k + B(w_k)u_k$. We consider the one-step decision-making process with a quadratic cost on state and action: $C := u_0^T R u_0 + x_1^T Q x_1$, where $x_1 = A(w_0)x_0 + B(w_0)u_0$. Here, $R \succ 0$ and $Q \succeq 0$. We consider a 10-dimensional state space with a 5-dimensional control input space. The number of realizations is taken to be $L = 3$ for ease of visualization. The L different $A(w_k)$ and $B(w_k)$ matrices corresponding to each realization are generated randomly by independently sampling elements of the matrices from the standard normal distribution $\mathcal{N}(0, 1)$. The cost matrix Q is a randomly generated positive semi-definite matrix and R is the identity. The initial states x^* were drawn randomly from the standard normal distribution $\mathcal{N}(0, I)$ where I denotes the identity matrix. The true envelope was generated by taking the convex hull of a set of random samples in the probability simplex Δ^L.

Figure 4 shows the outer approximations of the risk envelope obtained using Algorithm 1. We observe rapid convergence (approximately 20 sampled states x^*) of the outer approximations $\mathcal{P}_o$ (red) to the true risk envelope $\mathcal{P}$ (green). Figure 5 shows the mean squared error (on an independent test set with 30 demonstrations) between actions predicted using the sequentially refined polytope approximations generated by Algorithm 1 and the expert's true actions, as a function of the number of training demonstrations. One can observe rapid convergence in prediction performance after just 10 demonstration samples, further highlighting the data efficiency of the proposed algorithm.



(a) 5 data points

(b) 10 data points

(c) 15 data points

(d) 20 data points

Figure 4: Rapid convergence of the outer approximation of the risk envelope.

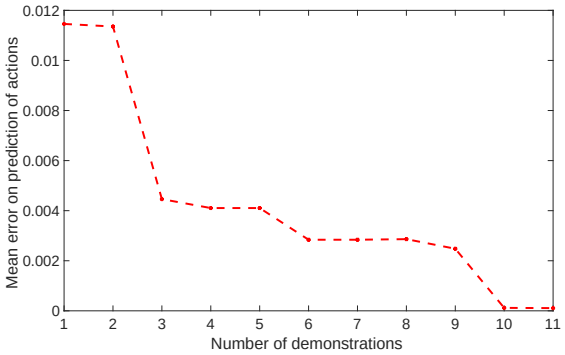


Figure 5: Rapid decrease of the mean squared error between predicted and expert's actions on an independent test set, as a function of the number of training demonstrations.

3.2 Unknown Cost Function

We next consider the more general case where both the expert's cost function $C(x_1, u_0)$ and risk measure are unknown. We parameterize the cost function as a linearly weighted combination of cost features, i.e.,

$$C(x_1, u_0) = c^T \phi(x_1, u_0),$$

where $c \in \mathbb{R}^H_{\geq 0}$ is a vector of *unknown* weights and $\phi : \mathbb{R}^n \times \mathcal{U} \to \mathbb{R}^H$ denotes the cost feature mapping from state and control input to an $H-$dimensional real vector. Since the solution of problem (6) solved by the expert is invariant to (i) constant shifts (Axiom A2 in Definition 1) and one can thus absorb negative signs into the cost features, and (ii) positive scalings (Axiom A3 in Definition 1), one can assume without loss of generality that the feature weights c are nonnegative and sum to one. Extending the notation in (7), let $\phi^{[j]}$ denote the feature vector when disturbance $w_0 = w^{[j]}$ is realized so that

$$g(x_0, u_0)(j) = c^T \phi^{[j]}(x_0, u_0), \quad j = 1, \ldots, L. \tag{17}$$

Thus, problem (8) now takes the form:

$$\begin{aligned} \min_{u_0 \in \mathcal{U}, \tau} \quad & \tau \\ s.t. \quad & \tau \geq \sum_{j=1}^{L} \sum_{h=1}^{H} v_i(j) c(h)\, \phi_h^{[j]}(x_0, u_0), \quad i \in \{1, \ldots, N_V\}. \end{aligned}$$

With this cost structure, we see that the KKT conditions derived in Section 3.1 now involve *products* of the feature weights c and the vertices v_i of $\mathcal{P}$. Thus, an analogous version of optimization problem (9) can be used to bound the optimal value. This problem will now contain products of the feature weights c and probability vector v. The key idea here is to introduce new *matrix* decision variables z that replace each product $v(j)c(h)$ by a new variable z_{jh} which allows us to re-write problem (9) as an LP in (z, σ_+, σ_-), with the addition of the following two simple constraints: $0 \leq z_{jh} \leq 1$, for all j, h, and $\sum_{j,h} z_{jh} = 1$. In a manner analogous to Theorem 2, this optimization problem allows us to obtain bounding hyperplanes *in the space of product variables* z which can then be aggregated as in Algorithm 1. Denoting this polytope as $\mathcal{P}_z$, we can then proceed to solve the "forward" problem (i.e., computing actions at a given state x) by solving the following optimization problem:

$$\min_{u_0 \in \mathcal{U}} \max_{z \in \mathcal{P}_z} \sum_{j,h} z_{jh} \phi_h^{[j]}(x_0, u_0). \tag{18}$$

This problem can be solved by enumerating the vertices of the polytope $\mathcal{P}_z$ in a manner similar to problem (8). Similar to the case where the cost function is known, this provides us with a way to conservatively approximate the expert's decision-making process (in the sense that we are considering a larger risk envelope).

3.2.1 Approximate Recovery of Cost and Risk Measure

While the procedure described above operates in the space of product variables z and does not require explicitly recovering the cost function and risk envelope separately, it may nevertheless be useful to do so for two reasons. First, the number of vertices of $\mathcal{P}_z$ may be quite large (since the

space of product variables may be high dimensional) and thus solving the forward problem (18) may be computationally expensive. Recovering the cost and risk envelope separately allows us to solve a smaller optimization problem (since the risk envelope is lower dimensional in this case). Second, recovering the cost and risk measure separately may provide additional intuition and insights into the expert's decision-making process and may also allow us to make useful predictions in novel settings (e.g., where we expect the expert's risk measure to be the same but not the cost function or vice versa).

Here we describe a procedure for approximately recovering the feature weights and the risk envelope from the polytope $\mathcal{P}_z$. The key observation that makes this possible is to note that the matrix z containing the variables z_{jh} is equal to the outer product vc^T by definition. Hence, for $h = 1, \ldots, H$, we have:

$$\sum_{j=1}^{L} z_{jh} = \sum_{j=1}^{L} v(j)c(h) = c(h) \sum_{j=1}^{L} v(j) = c(h). \tag{19}$$

The last equality follows from the fact v is a probability vector and sums to 1. Similarly, for $j = 1, \ldots, L$, we have:

$$\sum_{h=1}^{H} z_{jh} = \sum_{h=1}^{H} v(j)c(h) = v(j) \sum_{h=1}^{H} c(h) = v(j). \tag{20}$$

The last equality follows from the fact that we assumed without loss of generality that the feature weights sum to 1.

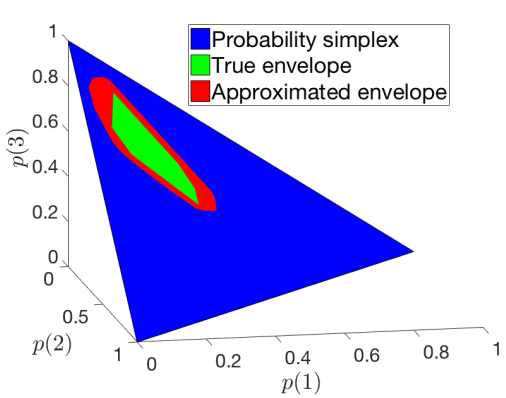
Let $\{\hat{z}_i\}$ be the set of (matrix-valued) vertices of the polytope $\mathcal{P}_z$. Then, by applying equations (19) and (20) to each vertex $\hat{z}_i$, we obtain a set of estimates of the feature weight vector c and a set of vectors in the probability simplex Δ^L, the convex hull of which gives an approximation of the risk envelope $\mathcal{P}$. If we have exactly recovered the polytope $\mathcal{P}_z$ in the space of product variables and the vertices $\hat{z}_i$ each have rank one, then it follows from problem (18) that the feature weight estimates will coincide and the convex hull of the probability vectors extracted from the vertices $\hat{z}_i$ will match the true risk envelope $\mathcal{P}$. In general however, this will not be the case since (i) there is no guarantee of exactly recovering the product polytope $\mathcal{P}_z$ (similar to how there is no guarantee of recovering the true risk envelope $\mathcal{P}$ in Algorithm 1), and (ii) $z = vc^T$ is a non-convex rank constraint that is not enforced in the KKT-based LP.

In light of these limitations, it is important to be able to gauge the quality of the estimates we obtain from the procedure above. We can do this in two ways. First, if the estimates of the weight vector are tightly clustered, this is a good indication that we have an accurate recovery. Second, if each vertex $\hat{z}_i$ of the polytope is close to a rank one matrix, then this is again a good indication (since the true product variables z equal vc^T).

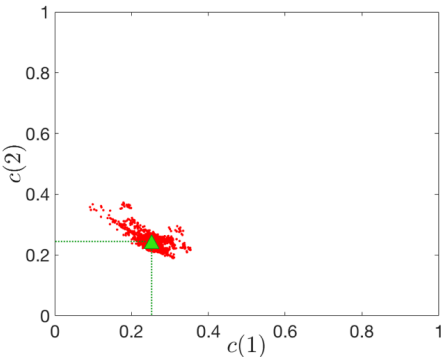
3.2.2 Example: Linear-Quadratic System

Consider the same system as in Section 3.1.1, but now we assume that the cost function is unknown. We take the cost function as the weighted sum of three quadratic features (i.e., $H = 3$). The quadratic features are generated randomly by taking them to be equal to SS^T, where the elements of S are sampled from the standard normal distribution. The corresponding weights are drawn uniformly between 0 and 1 and are normalized to sum to 1.

Figure 6 a) illustrates the tightness of the approximate envelope as compared with the true polytope, while Figure 6 b) is a scatter plot of the first two feature weights (the third is uniquely determined given the first two) as recovered from applying eq. (19) to each vertex of the compound polytope $\mathcal{P}_z$. Notice that the cost feature weight estimates are tightly clustered near the true weights.



(a) Polytope estimate.



(b) Feature weight estimates (red), true weights marked by green triangle.

Figure 6: Approximated risk envelope and cost feature weights from 200 state-control pair demonstrations.

4 Risk-sensitive IRL: Multi-step case

We now generalize the one-step decision problem to the multi-step setting. We consider a model where the disturbance w_k is sampled every $N > 1$ time-steps and held constant in the interim. Such a model generalizes settings where disturbances are sampled i.i.d. at every time-step (corresponding to $N = 1$ in our model) and it allows us to model delays in the expert's reaction to changing disturbances. We model the expert as planning in a *receding horizon* manner by looking ahead for a finite horizon longer than N steps, executing the computed policy for N steps, and iterating. Owing to the need to account for future disturbances, the multi-step finite-horizon problem is a search over control *policies* (i.e., the executed control inputs depend on which disturbance is realized).

4.1 Prepare-React Model: Preliminaries

In this section we reprise the "prepare" – "react" model introduced in (Majumdar et al., 2017), and depicted below in Figure 7. The expert's policy is decomposed into two phases (shown in Figure 7), referred to as "prepare" and "react." Intuitively, this model captures the idea that in the period preceding a disturbance (i.e., the "prepare" phase) the expert controls the system to a state from which he/she can recover well (in the "react" phase) once a disturbance is realized. Studies showing that humans have a relatively short look-ahead horizon in uncertain decision-making settings lend credence to such a model (Carton et al., 2016). As in (Majumdar et al., 2017), the delay parameter n_d would be learned directly from the demonstrations. To account for nested re-planning stages, we need to define a notion of *dynamic* risk measures, used to assess risk over sequential realizations of uncertainty.

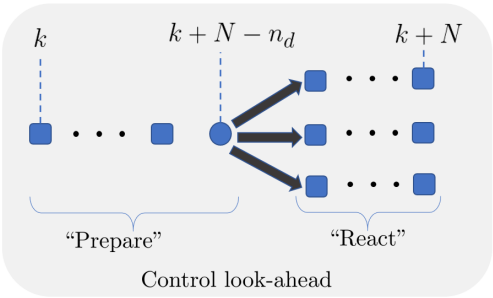


Figure 7: Scenario tree centered on a disturbance sampled at time $k+N-n_d$, where $N > n_d$. The "prepare" phase precedes each disturbance realization by $N-n_d$ steps, while the "react" phase follows it by n_d steps; in total, an individual planning stage last N steps. Note that during the prepare and react phases the dynamics are deterministic, as we assume that the disturbances stay constant for N steps in between sampling times. As we model the expert as a receding horizon planner, the expert's planning problem at time k would account for nested re-planning stages at time $k+N, k+2N, \ldots$ up to some look-ahead horizon. The expert would then execute their policy for the first N steps and then resolve the planning problem at time $k+N$ with a receded horizon.

4.2 Dynamic Risk Measures

Consider a discrete-time stochastic cost sequence $\{Z_t\}$, where $Z_t \in \mathcal{Z}_t$ the space of real-valued random variables at stage t. Let $\mathcal{Z}_{t:t'} := \mathcal{Z}_t \times \cdots \times \mathcal{Z}_{t'}$ where $t < t'$. A *dynamic risk measure* is a *sequence* of risk measures $\rho_{t:t'} : \mathcal{Z}_{t:t'} \to \mathcal{Z}_t, t = 0, \ldots, t'$, each mapping a future stream of random costs into a risk assessment at stage t and satisfying the monotonicity property $\rho_{t:t'}(Z_{t:t'}) \leq \rho_{t:t'}(Y_{t:t'})$ for all $Z_{t:t'}, Y_{t:t'} \in \mathcal{Z}_{t:t'}$ such that $Z_{t:t'} \leq Y_{t:t'}$. The monotonicity property is an intuitive extension of the monotonicity property for single-step risk assessments, and an arguably defensible axiom for all risk assessments.

To give dynamic risk measures a concrete functional form, we need to generalize the CRM axioms presented in Definition 1 to the dynamic case.

Definition 2 (Coherent One-Step Conditional Risk Measures)**.** *A* coherent one-step conditional risk measure *is a mapping $\rho_t : \mathcal{Z}_{t+1} \to \mathcal{Z}_t$, for all $t \in \mathbb{N}$, that obeys the following four axioms. For all $Z_{t+1}, Y_{t+1} \in \mathcal{Z}_{t+1}$ and $Z_t \in \mathcal{Z}_t$:*

A1. Monotonicity: $Z_{t+1} \leq Y_{t+1} \Rightarrow \rho_t(Z_{t+1}) \leq \rho_t(Y_{t+1})$.

A2. Translation invariance: $\rho_t(Z_{t+1} + Z_t) = \rho_t(Z_{t+1}) + Z_t$.

A3. Positive homogeneity: $\forall \lambda \geq 0,\ \rho_t(\lambda Z_{t+1}) = \lambda \rho_t(Z_{t+1})$.

A4. Subadditivity: $\rho_t(Z_{t+1} + Y_{t+1}) \leq \rho_t(Z_{t+1}) + \rho_t(Y_{t+1})$.

Note that each ρ_t is a random variable on the space $\mathcal{Z}_t$ and given the discrete underlying probability space, each component of ρ_t is uniquely identified by the sequence of disturbances preceding stage t (hence the term *conditional*). Furthermore, it is readily observed that a mapping $\rho_t : \mathcal{Z}_{t+1} \to \mathcal{Z}_t$ is a coherent one-step conditional risk measure if and only if each component of ρ_t is a CRM.

As investigated in (Ruszczyński, 2010), in order for dynamic risk assessments to satisfy the intuitive monotonicity condition and to ensure rationality of evaluations over time, a dynamic risk measure must have a compositional form:

$$\begin{aligned} \rho_{t:t'}(Z_{t:t'}) :=& Z_t + \rho_t\left(Z_{t+1} + \rho_{t+1}\left(Z_{t+2} + \ldots + \rho_{t'-1}(Z_{t'})\cdots\right)\right) \\ =& \rho_t \circ \cdots \circ \rho_{t'-1}(Z_t + \cdots + Z_{t'}), \end{aligned} \tag{21}$$

where each ρ_t is a coherent one-step conditional risk measure, and the second equality follows by the translational invariance property. Figure 8 provides a helpful visualization of such a compounded functional form.

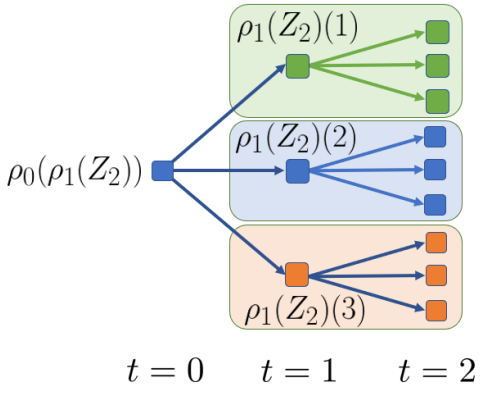


Figure 8: A scenario tree with three uncertain outcomes at each stage. The one-step risk mapping $\rho_1(Z_2) \in \mathcal{Z}_1$ maps the random cost $Z_2 \in \mathcal{Z}_2$ to a risk assessment at stage 1, i.e., is a random variable on $\mathcal{Z}_1$ and is thus isomorphic to the space $\mathbb{R}^3_{\geq 0}$. Here, each component j of $\rho_1(Z_2)$, i.e., $\rho_1(Z_2)(j)$, associated with node j at stage 1 (e.g., for $j = 1$, the green node), is a CRM over the children of node j at stage 2. The mapping $\rho_0(\rho_1(Z_2))$ subsequently maps the risk-assessments at stage 1 (i.e., $\rho_1(\cdot)$) back to stage 0.

4.3 Prepare-React Model: Formal Definition

We are now ready to formally define the expert's multi-step problem, from the perspective of time-step k, with look-ahead horizon $T\,N$ steps, where $T \in \mathbb{N}_{\geq 1}$ denotes the number of *branching events* (i.e., disturbance samples) within the prediction horizon. According to the prepare-react model introduced earlier, we assume that the disturbance mode for the first $N - n_d$ steps starting at time-step k corresponds to w_{k-n_d} (i.e., the disturbance mode realized at the last sampling event), following which the disturbance is re-sampled every N steps. An illustration of the nested prepare-react model with a look-ahead horizon $T = 2$ is provided in Figure 9.

Let $x_{k'|k}$ denote the predicted state for time-step $k + k'$, where $k' \in [0,\, TN - 1]$, as predicted at time-step k within the expert's multi-step optimization problem. Similarly, let $\{w'_t\}, t \in [0, T-1]$ represent the predicted disturbance sequence, where each $w'_t \in \{1, \ldots, L\}$. Let $\hat{\pi}_t(\boldsymbol{\omega}_{t-1}, w'_t),\ t \in [0, T-1]$ denote the expert's *"prepare" – "react" control policy* for stage t (corresponding to time-steps $k' \in [tN,\, (t+1)N - 1]$), where we make explicit the dependency on the partial predicted disturbance history $\boldsymbol{\omega}_{t-1} := \{w'_{-1}, w'_0, \ldots, w'_{t-1}\}$ and the next predicted disturbance mode w'_t, while we omit in the interest of brevity the dependency on x_k and partial policy history $\{\hat{\pi}_0, \ldots, \hat{\pi}_{t-1}\}$. We take $w'_{-1} =: w^*_{-1|k}$ to represent the actual disturbance mode in progress at the time of solving the multi-stage optimization problem at time-step k. Note that, by causality, only the "react" portion of $\hat{\pi}_t$ may be a function of w'_t but not the "prepare" portion. Finally, let $C_{tN:(t+1)N-1}(x_{tN|k}, \hat{\pi}_t(\cdot))$ denote the accumulated cost (i.e., a random variable adapted to the filtration generated by the sequence $\{w'_t\}$) over time-steps $k' \in [tN, (t+1)N - 1]$ given the "prepare" – "react" control policy $\hat{\pi}_t$ at stage t, that is,

$$C_{tN:(t+1)N-1}(x_{tN|k}, \hat{\pi}_t(\cdot)) = \sum_{k'=tN}^{(t+1)N-1} C(x_{k'|k}, \hat{\pi}_t(x_{k'|k})).$$

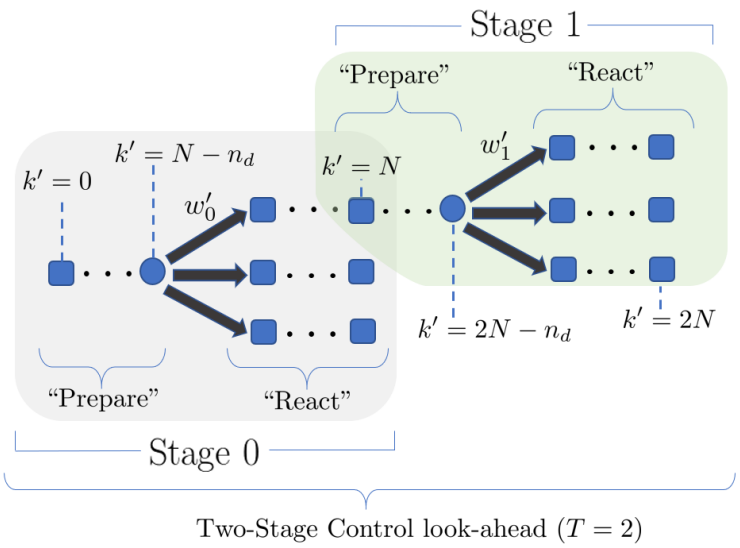


Figure 9: Multi-stage scenario tree for the prepare-react multi-step problem at time k (indexed internally using k' for simplicity). The disturbance is sampled every N steps. The control look-ahead consists of multiple nested branches of "prepare" and "react" sequences (indexed as "stages" in the figure above); shaded green in the figure above is one such nested branch corresponding to $w'_0 = w^{[1]}$. To evaluate costs over stage 1, it is assumed that the expert is using the conditional CRM $\rho_1(\cdot)$ where for each realization of $x_{N|k}$ (identified uniquely by the observed disturbance branch at stage 0), the expert uses the static CRM $\rho(\cdot)$ over the nested outcomes (shown in green for one possible realization of $x_{N|k}$). The observed control sequence is the beginning "prepare" – "react" sequence corresponding to the actual realized disturbance $w^*_{0|k}$.

As in Section 3.2, we assume that the time-step cost $C(x, u)$ is represented by a linear combination $c^T\phi(x, u)$ of features $\phi(x, u)$. The expert's multi-step optimization problem is then given as:

$$\min_{\substack{\hat{\pi}_t \\ t\in[0,T-1]}} \rho_0\Big(C_{0:N-1}(\cdot,\hat{\pi}_0) + \rho_1\big(C_{N:2N-1}(\cdot,\hat{\pi}_1) + \cdots + \rho_{T-1}(C_{(T-1)N:TN-1}(\cdot,\hat{\pi}_{T-1}))\big)\cdots\Big), \quad (22)$$

where each ρ_t, $t = 0, \ldots, T-1$ is a coherent one-step *conditional* risk measure such that each *component* of ρ_t is a CRM $\rho(\cdot)$ with respect to the probability space $(\mathcal{W}, 2^{\mathcal{W}}, p)$ and characterized by *the fixed risk envelope* $\mathcal{P} \subseteq \Delta^L$. Leveraging the translational invariance property, the objective may be equivalently re-written as

$$\begin{aligned}&\varrho(C_{0:TN-1}) := \rho_0 \circ \cdots \circ \rho_{T-1}(C_{0:TN-1})\\ =&C_{0:N-n_d} + \rho_0\Big(C_{N-n_d+1:N-1} + C_{N:2N-n_d} + \rho_1\big(C_{2N-n_d+1:2N-1} + \cdots + \rho_{T-1}\left(C_{TN-n_d+1:TN-1}\right)\cdots\big)\Big).\end{aligned}$$

One should notice that (1) for each $t \in \{0, \ldots, T-1\}$, the cost sequence $C_{tN:(t+1)N-1}$ is split across the risk operator ρ_t due to the "prepare"–"react" structure and the translational invariance property, (2) the risk mapping is over accumulated costs, i.e., in the notation of eq. (21), the stage t random cost Z_t corresponds to the accumulated cost $C_{tN:(t+1)N-1}$ since disturbances are sampled every N steps, and (3) since problem (22) is solved in receding horizon fashion and thus x_k is *known* at time k, $\varrho(\cdot)$ is a real valued *function*. The observed input from the expert is the first stage optimal "prepare" – "react" control policy $\hat{\pi}^*_0(w^*_{-1|k}, w^*_{0|k})$ where $w^*_{0|k}$ represents the actual

disturbance mode sampled after time-step k, following which the expert re-solves the problem at time $k+N$, with a receded horizon up to time-step $k+N+TN$.

Notice that by setting $T=1$, one recovers the single-stage "prepare – react" model presented in (Majumdar et al., 2017). The strategy in (Majumdar et al., 2017) is to reduce the multi-step inference problem to a mathematically equivalent single-step problem by estimating the (un-observed) control policies of the human agent, corresponding to the *un-realized* disturbance branches. Specifically, consider the scenario tree decomposition in Figure 7. If disturbance $w^{[3]}$ is realized at time-step $k+N-n_d$, then we only observe the "react" control sequence corresponding to the third branch. The algorithm in (Majumdar et al., 2017) proceeded by first inferring the "react" control sequences for the un-observed branches and then constructing a bounding hyperplane using a similar version of problem (9). In a multiple-stage setting, however, it is exceedingly difficult to exactly infer (or approximate) the unobserved control policies as each of these policies involves an *unobserved* nested optimization over future branching events. Consequently, the optimality conditions of an observed control policy are defined by equalities that are non-linear in the unobserved variables. Therefore, extending the use of KKT conditions to infer an outer approximation of the risk envelope in the style of Theorem 2 leads to an *intractable* non-convex optimization problem. To address this fundamental observability issue, we introduce a *semi-parametric* representation of the CRM, discussed next.

4.3.1 Semi-Parametric CRM

Fix a set of M normal vectors $a_j \in \mathbb{R}^L, j=1,\dots,M$. Let $\mathcal{P}_r$ denote the polytope defined by the halfspace constraints

$$\mathcal{P}_r := \{v \in \Delta^L \mid a_j^T v \leq b(j) - r(j), \quad j=1,\dots,M\}, \tag{23}$$

where for each j, $b(j) := \max_{v\in\Delta^L} a_j^T v$, and r is a parameter vector in $\mathbb{R}^M$. The CRM with risk envelope $\mathcal{P}_r$ is denoted as $\rho^r(\cdot)$; explicitly,

$$\rho^r(Z) = \max_{v\in\mathcal{P}_r} \mathbb{E}_v[Z], \tag{24}$$

where $Z \in \mathbb{R}^L$ is a discrete random variable with L possible realizations (see Figure 10).

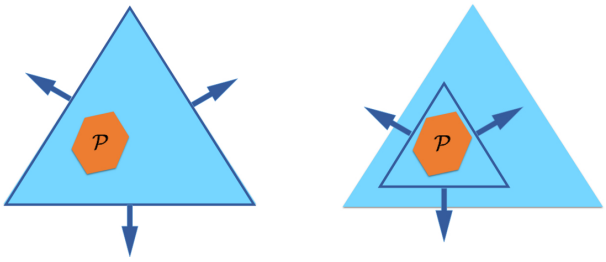


Figure 10: Schematic illustration of a semi-parametric CRM for a 3-scenario outcome space. The true risk envelope $\mathcal{P}$ is shown in orange; the boundary of the approximation polytope $\mathcal{P}_r$ is shown in dark blue. Left: the polytope $\mathcal{P}_r$ with $r=0$. Right: the polytope $\mathcal{P}_r$ for some $r \in \mathbb{R}^M_{>0}$. The arrows denote the a priori fixed normal vectors $\{a_j\}_{j=1}^3$.

This induced CRM is termed *semi*-parametric since unlike methods where one seeks to find the parameters defining a *fixed* disutility function (e.g., Shen et al. (2014); Ratliff and Mazumdar

(2017)), here we *do not* assume a fixed chosen risk measure. Instead, by parameterizing the risk measure in the dual space (via its risk envelope characterization), we retain the generality to recover any polytopic CRM, given a sufficient number of normal vectors a_j. A potential method to choose the normal vectors a_j is to take the halfplane normals from the multi-step KKT method described in (Majumdar et al., 2017).

In order to ensure that the polytope $\mathcal{P}_r$ defined in eq. (23) is non-empty, we define the extended polytope

$$\tilde{\mathcal{P}}_r := \{(v, r) \in \Delta^L \times \mathbb{R}^M \mid a_j^T v + r(j) \leq b(j), \quad j = 1, \ldots, M\}.$$

Define $\mathcal{R} := \text{proj}_r \tilde{\mathcal{P}}_r$ as the projection of polytope $\tilde{\mathcal{P}}_r$ along the r variables. Then, $r \in \mathcal{R}$ ensures that the polytope $\mathcal{P}_r$ is non-empty. It is readily observed that the set $\mathcal{R}$ is also a polytope.

4.3.2 Constrained Maximum Likelihood

Given the semi-parametric representation of the CRM in (24), the RS-IRL problem reduces to inference over the offset vector r, and cost feature weight vector c. We will perform this inference using a constrained maximum likelihood model. Consider, first, a likelihood model inspired by the MaxEnt IRL framework (Ziebart et al., 2008) where we assume

$$\Pr(\hat{\pi}_0(w^*_{-1|k}, w^*_{0|k})) \propto \exp\left(-\tau[w^*_{-1|k}, w^*_{0|k}]\right), \tag{25}$$

where, $\tau[w^*_{-1|k}, w^*_{0|k}]$ is the optimal value of (22), computed using the semi-parametric CRM defined in (24) and *conditioned* on $w'_0 = w^*_{0|k}$ and $\hat{\pi}_0 = \hat{\pi}_0(w^*_{-1|k}, w^*_{0|k})$ (see Appendix B for a detailed derivation of this distribution). While the original MaxEnt IRL model is motivated by finding the maximum entropy distribution subject to an expected feature matching constraint, the robust performance of MaxEnt IRL even in the absence of such a statistical motivation has been extensively observed and leveraged in the IRL literature, particularly in the context of noisy or suboptimal demonstrations.

A key limitation of the MaxEnt model, however, lies in the complexity of estimating the partition function (normalization factor for the distribution in eq. (25)) and its gradients. The likelihood model in eq. (25) represents a distribution over all possible N-length *policies*. This makes sampling-based approximations intractable, as similarly observed in (Kretzschmar et al., 2016), and Laplace integral-based approximations as used in (Levine and Koltun, 2012) too imprecise.

In order to construct a tractable algorithm, we employ the simplification whereby at the beginning of any "prepare" stage (see Figure 9), the expert can only choose an *open-loop* control *trajectory* $\hat{u}$ of length N from a finite set of such trajectories $\prod$, thereby eliminating the notion of a "react" *policy* and replacing it with an open-loop sequence spanning the entire "prepare" – "react" stage. The set $\prod$ of these trajectories can be chosen for instance by running the K-Means clustering algorithm on the raw input trajectories. This simplification allows us to interpret problem (22) as a game between the expert with action set $\prod$ and nature with action set $\mathcal{W}$, and uniquely identify any predicted state $x_{tN|k}, t \in [1, T-1]$ using the predicted game history, i.e., disturbance history $\boldsymbol{\omega}_{t-1}$ and control history $\boldsymbol{u}_{t-1} := \{\hat{u}_0, \ldots, \hat{u}_{t-1}\}$. Leveraging such a discrete representation and dynamic programming, one can then construct the optimal solution to the expert's multi-stage optimization problem using a "risk-sensitive" Bellman recursion, defined below.

Terminal Stage: For all possible game histories at stage $T-1$, define

$$\begin{aligned}\tau[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}](\hat{u}) &:= \rho\left(C_{(T-1)N:TN-1}(x_{(T-1)N|k}, \hat{u})\right)\\ \hat{\pi}^*_{T-1}[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}] &:= \operatorname*{argmin}_{\hat{u}\in\prod} \tau[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}](\hat{u}).\end{aligned}$$

Recursion: For all possible game histories at stage t, for $t = T-2, \ldots, 1$:

$$\begin{aligned}\tau[\boldsymbol{u}_{t-1}, \boldsymbol{\omega}_{t-1}](\hat{u}) &:= \rho\left(C_{tN:(t+1)N-1}(x_{tN|k}, \hat{u}) + \min_{\hat{u}'\in\prod} \tau[\{\boldsymbol{u}_{t-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t-1}, w'_t\}](\hat{u}')\right)\\ \hat{\pi}^*_t[\boldsymbol{u}_{t-1}, \boldsymbol{\omega}_{t-1}] &:= \operatorname*{argmin}_{\hat{u}\in\prod} \tau[\boldsymbol{u}_{t-1}, \boldsymbol{\omega}_{t-1}](\hat{u}).\end{aligned}$$

First Stage:

$$\begin{aligned}\tau[w^*_{-1|k}](\hat{u}) &:= \rho\left(C_{0:N-1}(x_k, \hat{u}) + \min_{\hat{u}'\in\prod} \tau[\{\hat{u}\}, \{w^*_{-1|k}, w'_0\}](\hat{u}')\right)\\ \hat{\pi}^*_0[w^*_{-1|k}] &:= \operatorname*{argmin}_{\hat{u}\in\prod} \tau[w^*_{-1|k}]](\hat{u}).\end{aligned}$$

The value $\min_{\hat{u}\in\prod} \tau[w^*_{-1|k}](\hat{u})$ is the optimal value of problem (22). In the equations above, it is understood that for each $t \in [0, T-1]$, the accumulated cost $C_{tN:(t+1)N-1}$ is evaluated based on the previous disturbance mode w'_{t-1} for the first $N - n_d$ steps, followed by w'_t for the remaining n_d steps.

Given the structure of the optimal solution of problem (22), presented in Bellman form above using the true CRM $\rho(\cdot)$, we now construct a *computationally tractable* likelihood model for the parameters r and c by defining the *soft* risk-sensitive Bellman recursion using the semi-parametric CRM $\rho^r(\cdot)$. For the terminal stage, define

$$\tilde{\tau}[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}](\hat{u}) := \rho^r\left(C_{(T-1)N:TN-1}(x_{(T-1)N|k}, \hat{u})\right); \tag{26}$$

for all $t = T-2, \ldots, 1$, define:

$$\tilde{\tau}[\boldsymbol{u}_{t-1}, \boldsymbol{\omega}_{t-1}](\hat{u}) := \rho^r\left(C_{tN:(t+1)N-1}(x_{tN|k}, \hat{u}) + \operatorname*{softmin}_{\hat{u}'\in\prod} \tilde{\tau}[\{\boldsymbol{u}_{t-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t-1}, w'_t\}](\hat{u}')\right); \tag{27}$$

finally, for the first stage, define:

$$\tilde{\tau}[w^*_{-1|k}](\hat{u}) := \rho^r\left(C_{0:N-1}(x_k, \hat{u}) + \operatorname*{softmin}_{\hat{u}'\in\prod} \tilde{\tau}[\{\hat{u}\}, \{w^*_{-1|k}, w'_0\}](\hat{u}')\right), \tag{28}$$

where $\operatorname{softmin}_x f(x) := -\log \sum_x \exp(-f(x))$.

Let $\hat{u}^*_t$ be the closest (in $\mathcal{L}_2$ norm) trajectory in $\prod$ to the observed control sequence over time-steps $[tN, (t+1)N-1]$[2]. Similar to the MaxEnt IRL approach, we allow for imperfect human demonstrations by postulating that lower risk-sensitive cost actions (i.e., $\tilde{\tau}[w^*_{-1|tN}](\hat{u})$) are exponentially preferred, i.e.,

$$\Pr(\hat{u}) \propto \exp\left(-\beta\tilde{\tau}[w^*_{-1|tN}](\hat{u})\right), \tag{29}$$

[2]We use t here for notational consistency between the stage-wise decomposition of the multi-step problem and demonstrated action trajectories.

where $\beta > 0$ is an inverse temperature parameter. Thus, the likelihood of parameters r, c is given by:

$$l(r, c|\hat{u}_t^*) := \frac{\exp\left(-\beta\tilde{\tau}[w^*_{-1|tN}](\hat{u}_t^*)\right)}{\sum_{\hat{u}} \exp\left(-\beta\tilde{\tau}[w^*_{-1|tN}](\hat{u})\right)}. \tag{30}$$

As the expert is assumed to solve the problem in a receding horizon fashion, we may treat each $(w^*_{-1|tN}, \hat{u}_t^*)$ tuple in the demonstrated trajectory $\mathcal{T}^*$ independently. Consequently, the log likelihood given the entire trajectory is simply

$$l(r, c|\mathcal{T}^*) = \frac{1}{|\mathcal{T}^*|} \sum_{\hat{u}_t^* \in \mathcal{T}^*} -\beta\tilde{\tau}[w^*_{-1|tN}](\hat{u}_t^*) + \operatorname*{softmin}_{\hat{u}} \beta\tilde{\tau}[w^*_{-1|tN}](\hat{u}), \tag{31}$$

where $|\mathcal{T}^*|$ is the number of N-step demonstrations in the trajectory $\mathcal{T}^*$. The inference problem is then

$$\{r^*, c^*\} := \operatorname*{argmax}_{c \in \Delta^H,\ r \in \mathcal{R}} l(r, c|\mathcal{T}^*), \tag{32}$$

where, as before, we assume that the cost weights are non-negative and sum to one (and thus lie in the simplex Δ^H). We solve the problem using projected gradient descent on r and entropic mirror descent on c. The gradient formulas are derived by propagating gradients of the $\tilde{\tau}$ variables in recursive fashion from the terminal to the first stage (similar to the computation of $\tilde{\tau}$ itself) and leveraging LP sensitivity results. For ease of exposition, we provide these recursive formulas in Appendix C.

Remark 4. *While we lose the outer-approximation of the risk envelope and convergence guarantees associated with the KKT method, in its place we obtain a tractable algorithm that enables us to accommodate a substantially larger class of dynamic decision-making inference problems. Experimental results, as discussed in Section 5, confirm that the method works well in approximating a wide range of risk profiles.*

5 Example: Driving Game Scenario

In this section we apply our RS-IRL framework on a simulated driving game (Figure 1) with ten human participants to demonstrate that our approach is able to infer individuals' varying attitudes toward risk and mimic the resulting driving styles. In particular, we note that the experimental setting here constitutes a significantly more challenging and dynamic testbed than typical benchmark examples such as grid-world or sequential investment tasks.

5.1 Experimental Setting

The setting consists of a leader car and a follower car, simulated in the commercial driving simulator Vires VTD (VIRES Simulationstechnologie GmbH, 2017). Participants controlled the follower car with the Logitech G29 control suite, consisting of a steering wheel and pedals (Figure 1). The follower car is modeled using the simple car model with states: x_f (along-track position), y_f (lateral position), v_f (speed), θ_f (yaw angle) and δ_f (steering angle). The dynamics are given by:

$$\dot{x}_f = v_f \cos(\theta_f), \quad \dot{y}_f = v_f \sin(\theta_f), \quad \dot{v}_f = u_a, \quad \dot{\theta}_f = -\frac{v_f}{l}\tan(\delta_f), \quad \dot{\delta}_f = u_s, \tag{33}$$

where u_a and u_s are, respectively, the longitudinal acceleration and the steering rate inputs, and $l = 3.476$ m. The leader car plays the role of an "erratic driver" and is modeled with double integrator dynamics along-track and triple integrator dynamics in the lateral direction to mimic continuous steering inputs. The state of the leader's car is described by x_l (along-track position), y_l (lateral position), $v_{x,l}$ (forward speed), $v_{y,l}$ (lateral speed) and a_y (lateral acceleration). The dynamics are given by:

$$\dot{x} = v_x, \quad \dot{v}_x = w_x, \quad \dot{y} = v_y, \quad \dot{v}_y = a_y, \quad \dot{a}_y = w_y, \tag{34}$$

where $[w_x, w_y]^T$ is the leader's control input. We simulate this system in discrete time at 60 Hz and analyze the data with a time step $\Delta t = 0.1$ s.

In this setting, a disturbance $w^{[i]}$ corresponds to a sequence of control inputs executed by the leader car $w^{[i]} := \{(w_x, w_y)_k^{[i]}\}_{k=1}^N$ over N time steps. Each disturbance is sampled from a finite set $\mathcal{W} = \{w^{[1]}, \ldots, w^{[L]}\}$ with $L = 4$. These "disturbance" realizations correspond to different maneuvers for the leader (doing nothing, accelerating, decelerating, and swapping lanes) and are generated randomly according to the pmf $p = [0.3, 0.3, 0.3, 0.1]$.

The disturbance is sampled every $N = 15$ time steps. Thus, the leader car can be viewed as executing a random *maneuver* every 1.5 seconds. The whole system is described by the state:

$$\xi := [x_f, y_f, \theta_f, v_f, \delta_f, x_l, v_{x,l}, y_l, v_{y,l}, a_{y,l}]^T. \tag{35}$$

Participants in the study were informed that their goal was to follow the leader car (described as an "erratic driver"), as closely as possible in the x and y directions, while staying behind the leader and avoiding any collision. The leader car's four maneuvers were described to participants, along with the fact that these sequences of actions are generated every 1.5 s, *independent* of (as opposed to an interactive game) the participant's actions and position.

The experimental protocol for each participant consisted of three phases. The first phase (~ 1 minute) was meant for the participant to familiarize themselves with the simulator. The second and third phases (one minute each) involved the leader car acting according to the model described above (with actions being sampled according to the pmf p). The data collected during the second phase was used to train the model and data collected during the third phase was used to test it (the second and third phase disturbance sequences were kept same for all participants).

Note that the pmf p is *not* shared with the participants. This experimental setting may thus be considered *ambiguous*. However, since participants are exposed to a training phase where they may build a mental model of disturbances, the setting may also be interpreted as one involving risk.

While the "game" setting is identical to the one introduced in our earlier work in (Majumdar et al., 2017), the use of non-linear dynamics and a realistic driving simulator as opposed to the first-order integrator MATLAB game in (Majumdar et al., 2017) lends the experiment more realism. All data, algorithm, and plotting code is made available at `https://github.com/StanfordASL/RSIRL`.

5.2 Modeling and Implementation

We modeled participants' behavior using the multi-stage "prepare"–"react" framework presented in Section 4 with the "prepare" phase starting 0.7 seconds before the leader's action is sampled. The "react" phase thus extends to 0.8 seconds after the disturbance. This parameter was chosen as being roughly reflective of observed participant behavior during the *training* phase. We use $T = 2$

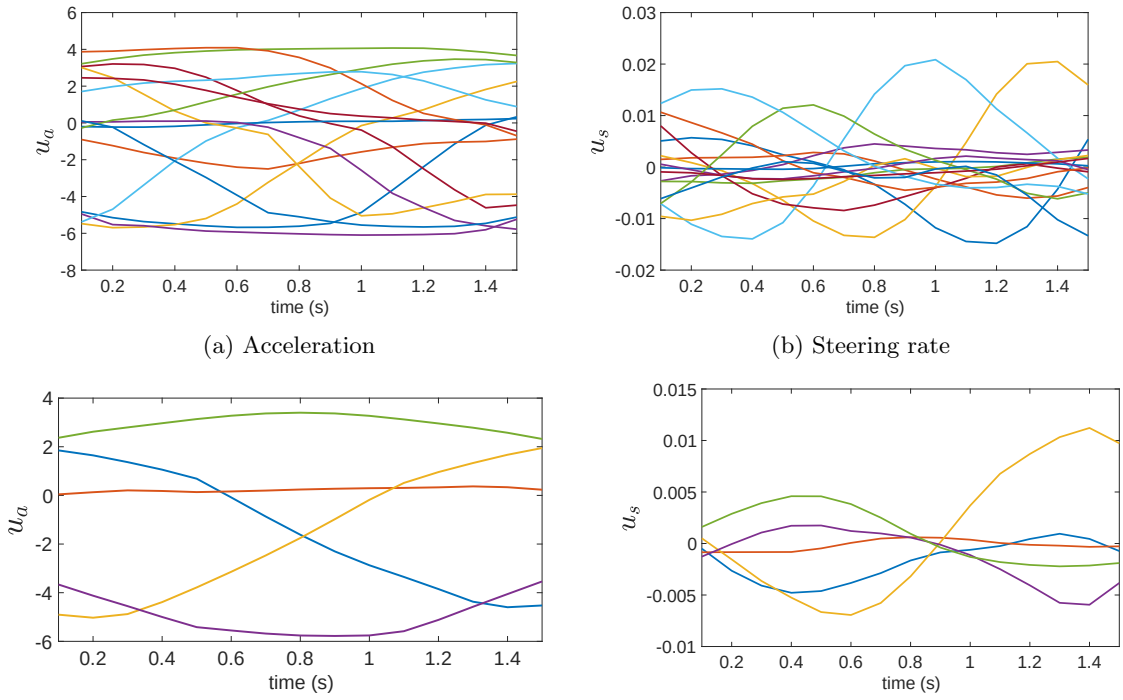
decision stages to model the participants. Hence, the planning horizon is $NT = 30$, which involves planning over two disturbance branching events in a receding horizon fashion.

We represent the cost function as a linear combination of the following features (with unknown weights):

- $\phi_1 = \mathbf{1}_{x_{\text{rel}}<2.5}[\log(1 + e^{-r_1(x_{\text{rel}}-2.5)}) - \log(2)]$,
- $\phi_2 = \mathbf{1}_{x_{\text{rel}}>2.5}[\log(1 + e^{r_2(x_{\text{rel}}-2.5)}) - \log(2)]$,
- $\phi_3 = \log(1 + e^{r_3|v_{x,\text{rel}}|}) - \log(2)$,
- $\phi_4 = r_4 \sum_{k=2}^{N}(u_{a,k} - u_{a,k-1})^2$,
- $\phi_5 = \log(1 + e^{r_5|y_{\text{rel}}|}) - \log(2)$,
- $\phi_6 = \mathbf{1}_{y_f>2}[\log(1 + e^{r_6(y_f-2)}) - \log(2)] + \mathbf{1}_{y_f<-2}[\log(1 + e^{-r_6(y_f+2)}) - \log(2)]$,

where x_{rel}, y_{rel}, and $v_{x,\text{rel}}$ are respectively the relative along-track position, lateral position, and along-track velocity between the leader and the follower. Hence, the first feature translates the instruction of staying behind the leader; the second, third, and fifth features penalize the relative distance and velocity between the leader and follower; the fourth feature penalizes change in longitudinal acceleration (effectively jerk of the trajectory); the sixth feature penalizes crossing the road boundaries. We use $r_1 = 1$, $r_2 = 0.05$, $r_3 = .1$, $r_4 = 1.0$, $r_5 = 0.1$, and $r_6 = 0.5$. These values were chosen to ensure that the costs were well conditioned over the usual range of relative states observed during the experiments.

In order to optimize the model by maximum likelihood estimation as described in Section 4, we discretized the control space $[u_a, u_s]$ to generate the participant action space. For each participant, we ran the K-Means clustering algorithm on the training control inputs, and chose 15 control trajectories for the first decision stage and 5 trajectories for the second stage. Since we model each participant as planning over a receding horizon with two decision stages, it is reasonable to assume that the plan for the second stage is not as fine-grained as over the first one. In addition to reducing the computational burden, this concept of mixing coarse-fine planning is a feature also described in (Carton et al., 2016) to model human locomotion. We observe that using 15 control trajectories for the first stage and 5 trajectories for the second stage was sufficient to generate a diverse expert action set in terms of accelerations (Figure 11a, 11c) and steering rates (Figure 11b, 11d); the span of all x/y traces resulting from the combination of these first and second stage control trajectories is shown in Figure 12.



(a) Acceleration

(b) Steering rate

(c) Acceleration

(d) Steering rate

Figure 11: Example of control trajectories computed by the K-Means algorithm using 15 centroids ((a), (b)) and 5 centroids ((c), (d)). Acceleration in m/s^2 and steering rate in rad/s.

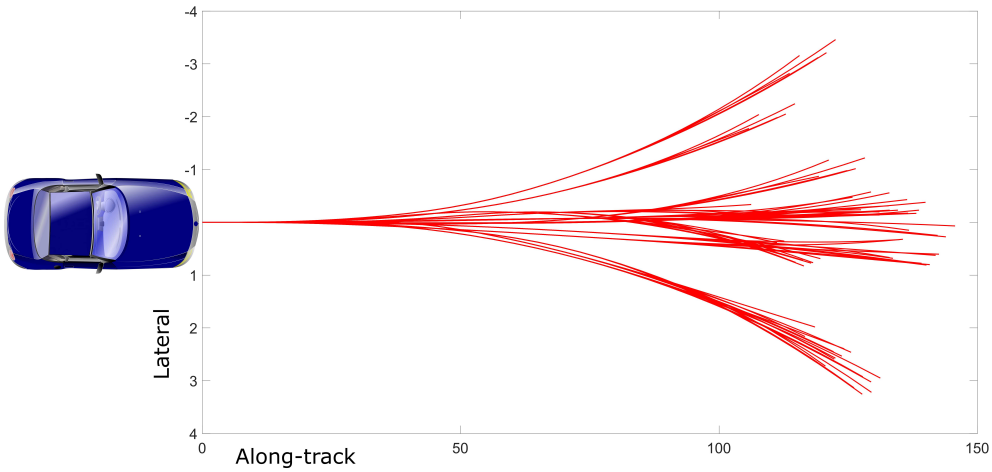


Figure 12: Span of all possible along-track/lateral (x/y) 3 second trajectories encapsulated within a single 2-stage optimization problem with the 15/5 discrete control trajectory space. All lengths in meters.

The polytope $\mathcal{P}_r$ (alternatively, the semi-parameterized CRM $\rho^r(\cdot)$) was parametrized with 8 normal vectors $\{a_j\}_{j=1}^8$ corresponding to the positive and negative standard basis vectors in $\mathbb{R}^4$ (i.e., $\{(e_i, -e_i)\}_{i=1}^4$). Our MATLAB implementation uses the parser YALMIP (Löfberg, 2004) and

the solver Mosek (ApS, 2017).

5.3 Results

Interestingly, our simulated driving scenario was rich enough to elicit a wide variety of qualitative behaviors from the ten participants. In particular, we observed two extreme policies. One extreme involved the driver following the leader very closely with a small separation (Figure 20). Another extreme was to follow the leader with a distance large enough to avoid any collision, often decelerating preemptively to avoid such an event (Figure 14). These two extremes can be interpreted as reflecting varying attitudes towards risk. The first policy corresponds to risk-neutral behavior, where the perceived (as captured by the inferred risk measure) probability of collision is lower than for highly risk-averse participants who were more sensitive to the worst-case eventuality (leader slowing down). We also observed a range of behaviors that lie between these two extremes.

We compare the RS-IRL approach with one where the expert is modeled as minimizing the expected value of his/her cost function computed with respect to the pmf p, in a receding horizon fashion with two decision stages. Similar to eq. (29), we assume that the risk-neutral stochastic policy is given by the Boltzmann distribution induced by the risk-neutral costs, thereby coinciding with the standard, MaxEnt IRL model and representing an important benchmark for comparison. We refer to this approach as risk-neutral IRL (RN-IRL). The analog of the recursion equations (26)–(28) for the risk-neutral model are obtained by simply replacing the conditional risk measures with the expected value with respect to the pmf p.

Since the expert is assumed to plan his/her decisions every 1.5 s in a receding horizon fashion, we evaluate RS-IRL and RN-IRL predictions based on their errors with respect to each 1.5 s observed demonstration in the test trajectory. In particular, we define the following error metric for predictions in $x_{\text{rel}} = x_l - x_f$:

$$\Delta x_{\text{rel},t} := \mathbb{E}\left[\sqrt{\sum_k (x^{\text{predicted}}_{\text{rel},k|t} - x^{\text{human}}_{\text{rel},k|t})^2}\right], \tag{36}$$

where $x^{\text{predicted}}_{\text{rel},k|t}$ and $x^{\text{human}}_{\text{rel},k|t}$ are, respectively, the predicted and actual x_{rel} trajectories at time $k \in [tN, (t+1)N]$ corresponding to the t^{th} 1.5 s segment in the demonstrated trajectory. The expectation is taken with respect to the stochastic policy (i.e., Boltzmann distribution) induced by the RS or RN costs (see eq. (29) and (51)). The errors Δy_{rel}, $\Delta v_{x,\text{rel}}$ and $\Delta v_{y,\text{rel}}$ are computed similarly. As RN-IRL consistently performed better with $T = 2$ decision stages, we only present comparison results between RS-IRL and RN-IRL for $T = 2$. To get a scale for the values reported in this section, the figure below illustrates the two cars almost colliding ($x_{\text{rel}} \approx 2.5$ m) and when they are 5 m apart.

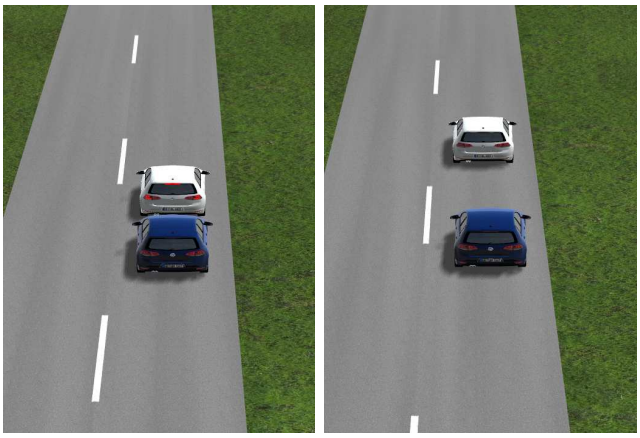


Figure 13: Left: Simulator visual when cars are almost at collision distance ($x_{\text{rel}} \approx 2.5$ m); Right: Simulator visual of a participant driving 5 meters behind the leader. Lane-width: 3 m.

5.3.1 Case Study # 1: Risk-Averse Participant

Figure 14 plots the x_{rel} trajectory (normalized by car length ≈ 4.2 m) during the third (test) phase for a participant inferred to be highly risk-averse. On average, the along-track relative distance is quite large (≈ 6 car-lengths). The expected prediction errors $\Delta x_{\text{rel},t}$ (normalized by car-length) from RS-IRL and RN-IRL for each of the 51 1.5 s demonstrations comprising the test phase are plotted in Figure 15a as absolute errors, and in Figure 15b as percentage differences with positive values indicating an improvement of RS-IRL over RN-IRL. A similar error plot is shown in Figure 15c and 15d for along-track velocity error $\Delta v_{x,\text{rel},t}$.

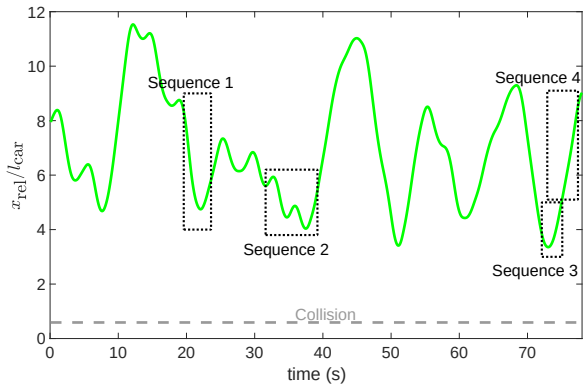
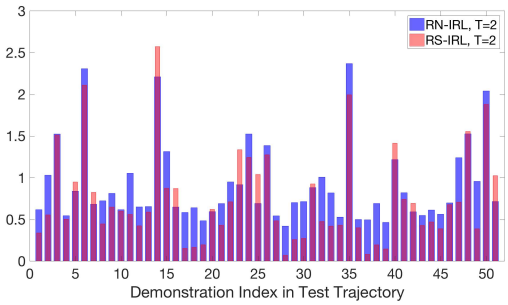
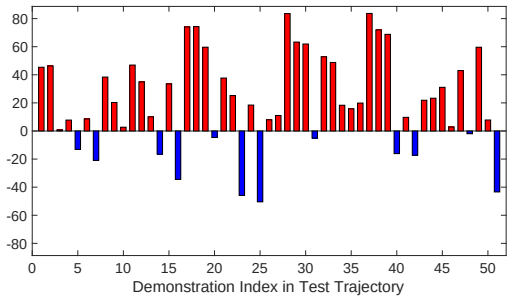


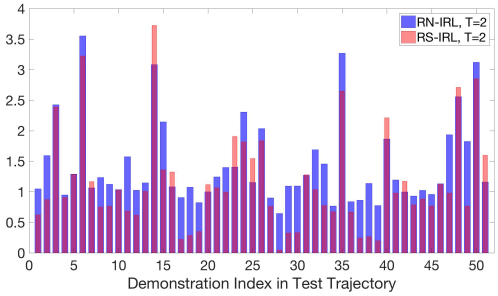
Figure 14: Full x_{rel} (longitudinal distance) trajectory (normalized by car length) for a highly risk-averse participant. On average, the relative distance is quite large (≈ 6 car-lengths). The boxed sections are discussed in further detail below.



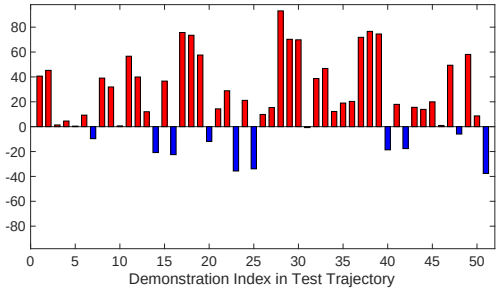
(a) Expected (w.r.t. stochastic policy) prediction errors $\Delta x_{\mathrm{rel},t}$ from RS-IRL and RN-IRL for each 1.5 s trajectory segment.



(b) Percentage improvement in $\Delta x_{\mathrm{rel},t}$ for the RS-IRL model over RN-IRL for each 1.5 s trajectory segment.

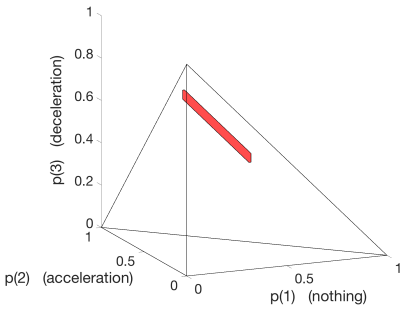


(c) Expected (w.r.t. stochastic policy) prediction errors $\Delta v_{x,\mathrm{rel},t}$ from RS-IRL and RN-IRL for each 1.5 s trajectory segment.

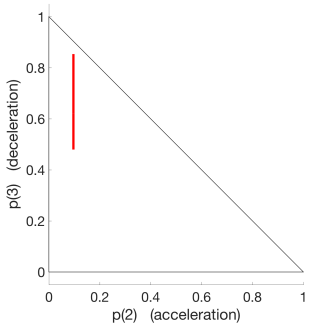


(d) Percentage improvement in $\Delta v_{x,\mathrm{rel},t}$ for the RS-IRL model over RN-IRL for each 1.5 s trajectory segment.

Figure 15: Comparison of the $\Delta x_{\mathrm{rel},t}$ and $\Delta v_{x,\mathrm{rel},t}$ prediction errors (normalized by car length) for the RS-IRL and RN-IRL models for a highly risk-averse participant. The RS-IRL model almost always outperforms RN-IRL, on average providing about 22% improvement.



(a) Projection of the risk-averse participant's risk envelope along the first three dimensions, corresponding to the {nothing, accelerate, decelerate} maneuvers by the leader.



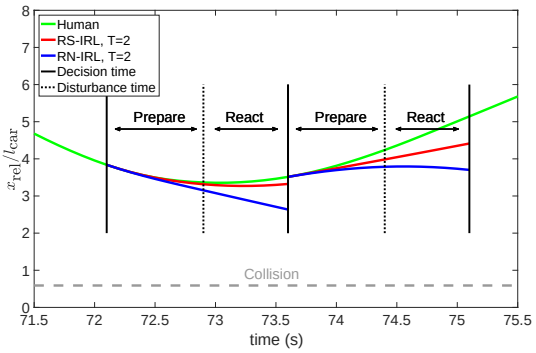
(b) Projection of the risk-averse participant's risk envelope along the second two dimensions, corresponding to the {accelerate, decelerate} maneuvers by the leader.

Figure 16: Inferred risk envelope for risk-averse participant. Notice the overweighting (and ambiguity therein) of probabilities associated with the leader's deceleration maneuver and the underweighting of the leader's acceleration. The wire-frame pyramid is the projection of the probability simplex Δ^4 onto the first three dimensions.

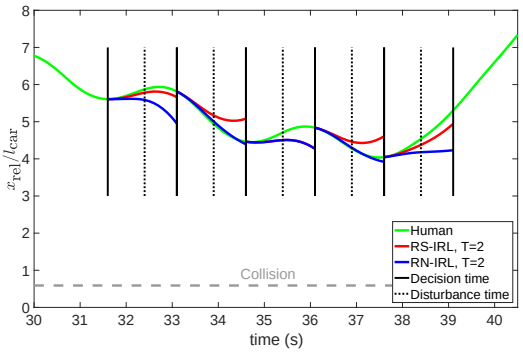
It can be observed that the RS-IRL model almost always outperforms RN-IRL, on average providing an improvement of about 22%. Notice however, the four prominent negative peaks in Figure 15b corresponding to the RN-IRL model outperforming RS-IRL on these instances. All four peaks can be explained by considering the inferred risk envelope $\mathcal{P}_r$, whose projections are plotted in Figures 16a and 16b. Notice how the participant's polytope is significantly biased towards high probabilities in the third dimension (corresponding to the decelerate maneuver for the leader), and furthermore, spans a wide range in probabilities ($\approx [0.5, 0.85]$) for this event. Thus, not only does the participant overweight the true probability of this outcome (0.3), he/she is also *ambiguous* about the true probability. On the other hand, the envelope suggests a very low and narrow range of probabilities for the leader's accelerate maneuver. These two properties are what result in the four negative peaks in Figure 15b.

Consider the first negative peak (corresponding to Sequence 1 in Figure 14). At this decision stage, the true distance decreases due to the participant accelerating. As the learned RS-IRL model assumes significant ambiguity in the leader's deceleration yet small probability in acceleration, the most probable trajectories from the RS-IRL model prefer weaker acceleration. Thus, this particular artifact may be attributed to a slight overfitting of the parameterized polytope offsets r. This overfitting manifests itself again at the very end (Sequence 4) where the cars are far apart (6 car-lengths) and the RS-IRL model predicts a weaker deceleration than that observed. Essentially the high bias associated with the leader's deceleration as encoded in the inferred polytope along with the already large separation between the cars (recall that the second feature penalizes large separations) leads to a less aggressive deceleration prediction. The remaining peaks (Sequence 2) are shown in detail in Figure 17b. At both these decision stages, the RS-IRL model *preempts*, by about one decision stage, a deceleration maneuver for the participant. Again this is a result of the deceleration ambiguity implied by the RS-IRL model and suggests that additional training data is needed to further refine (decrease) this ambiguity. This would naturally also alleviate the prediction errors in Sequences 1 and 4.

Notice however that all of the spurious predictions by the RS-IRL model as discussed above occur when the cars are quite far apart, on the order of 5–6 car-lengths. In contrast, when the two cars are quite close (e.g., less than 4 car-lengths during Sequence 3), there is now a significant collision risk in the event that the leader decelerates. Figure 17a illustrates the RS-IRL model correctly predicting the participant braking to increase the relative separation, and with high probability (≈ 0.89). In contrast, the RN-IRL induced stochastic policy is not only of high entropy but also suggests quite absurd trajectories that lead to a further decrease in the relative separation. Thus, these results suggest that when RS-IRL underperforms RN-IRL, it does so at non-critical stages characterized by low risk.



(a) Close-up of Sequence 3 in Figure 14 demonstrating the accuracy of RS-IRL over RN-IRL at a critical decision stage when cars are quite close.

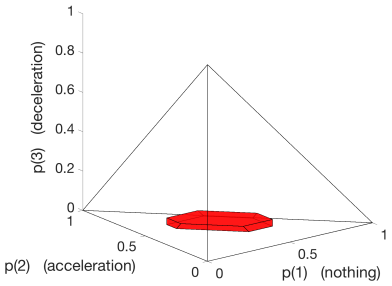


(b) Close-up of Sequence 2 in Figure 14 demonstrating the preemptive braking maneuvers predicted by RS-IRL. Notice that the error occurs at a non-critical decision stage when cars are far apart.

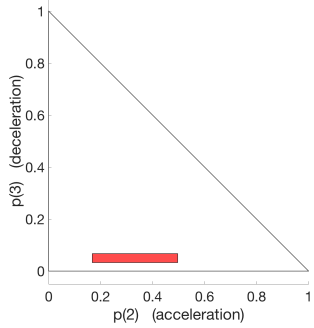
Figure 17: Comparisons of the *most-probable* (under the stochastic Boltzmann policy) RS-IRL and RN-IRL trajectory predictions in x_{rel} as compared with the true data.

5.3.2 Case Study # 2: Risk/Ambiguity-Averse Participant

We next discuss a participant for whom RS-IRL again significantly outperformed RN-IRL, however, due to a different manifestation of risk. Consider the inferred risk envelopes in Figure 18.



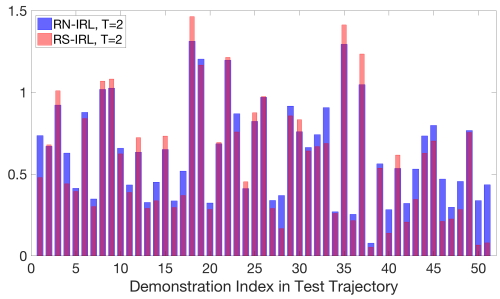
(a) Projection of polytope along the {nothing, accelerate, decelerate} "dimensions".



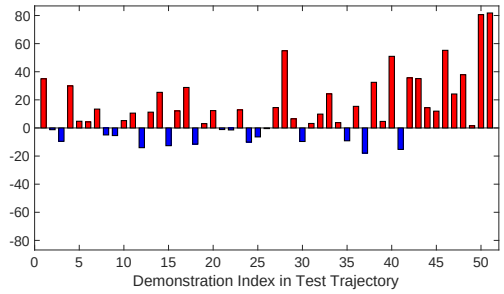
(b) Projection of polytope along the {accelerate, decelerate} "dimensions".

Figure 18: Inferred risk envelope for ambiguity-averse participant. Notice the drastic underweighting of probabilities associated with leader's deceleration and significant ambiguity regarding leader's acceleration.

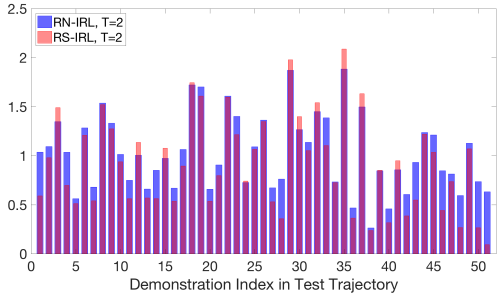
Notice that the participant places very low probability on deceleration yet is ambiguous regarding the leader's acceleration (i.e., a polar opposite of the first participant). This ambiguity led to several preemptive braking maneuvers in an attempt to maintain a "safe" distance to the leader car. Figures 19a–19d illustrate the consistency of RS-IRL's improvement over RN-IRL in predicting these maneuvers over the entire test trajectory.



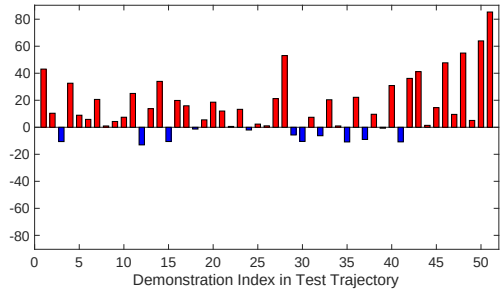
(a) Expected (w.r.t. stochastic policy) prediction errors $\Delta x_{\text{rel},t}$ from RS-IRL and RN-IRL for each 1.5 s trajectory segment.



(b) Percentage improvement in $\Delta x_{\text{rel},t}$ for the RS-IRL model over RN-IRL for each 1.5 s trajectory segment.



(c) Expected (w.r.t. stochastic policy) prediction errors $\Delta v_{x,\text{rel},t}$ from RS-IRL and RN-IRL for each 1.5 s trajectory segment.



(d) Percentage improvement in $\Delta v_{x,\text{rel},t}$ for the RS-IRL model over RN-IRL for each 1.5 s trajectory segment.

Figure 19: Comparison of the $\Delta x_{\text{rel},t}$ and $\Delta v_{x,\text{rel},t}$ prediction errors (normalized by car length) for the RS-IRL and RN-IRL models for an ambiguity-averse participant. The RS-IRL model almost always outperforms RN-IRL, on average providing about 13–14% improvement.

5.3.3 Case Study # 3: Risk-Neutral Participant

Finally, we consider a participant for whom both RS-IRL and RN-IRL performed on par with each other. Figure 20 plots the x_{rel} trajectory while Figure 21 illustrates the inferred risk envelope.

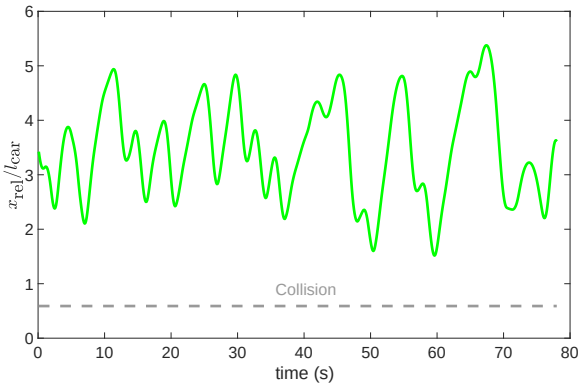
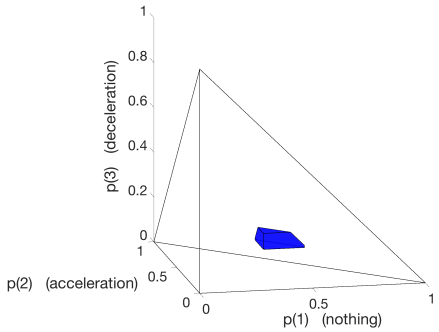
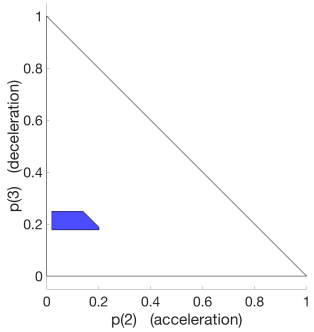


Figure 20: Full x_{rel} trajectory (normalized by car length) for a risk-neutral participant. On average, the relative distance is noticeably smaller, on the order of 3 car-lengths.



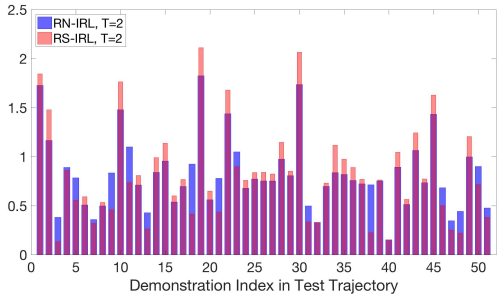
(a) Projection of polytope along the {nothing, accelerate, decelerate} "dimensions".



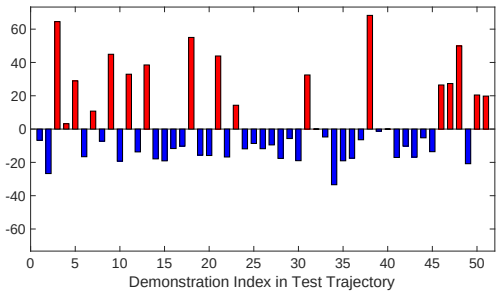
(b) Projection of polytope along the {accelerate, decelerate} "dimensions".

Figure 21: Inferred risk envelope for risk-neutral participant. Notice how there is no appreciable level of ambiguity nor is the polytope biased along any dimension; hence suggesting the risk-neutral categorization.

The inferred risk envelope features no bias along any dimension nor any appreciable level of ambiguity; thereby suggesting a risk-neutral profile for this participant. Furthermore, notice in Figure 20 that the participant stays quite a bit closer to the leader car than the first participant, a clear indicator of his/her risk-neutral stance. Figures 22 and 23 confirm the two models performing on par with each other. This, however, is to be expected for a strongly risk-neutral participant since the RS-IRL model subsumes the RN-IRL model.



(a) Expected (w.r.t. stochastic policy) prediction errors $\Delta x_{\mathrm{rel},t}$ from RS-IRL and RN-IRL for each 1.5 s trajectory segment.

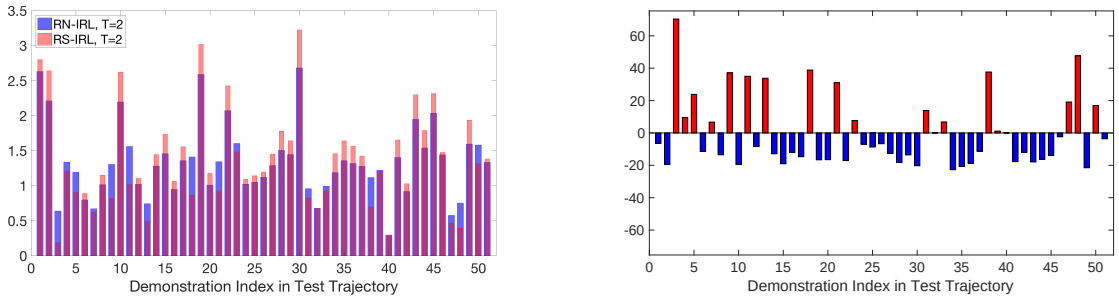


(b) Percentage improvement in $\Delta x_{\mathrm{rel},t}$ for the RS-IRL model over RN-IRL for each 1.5 s trajectory segment.

Figure 22: Comparison of the $\Delta x_{\mathrm{rel},t}$ prediction errors (normalized by car length) for the RS-IRL and RN-IRL models for a risk-neutral participant. The two models perform on par with each other.

5.3.4 Limitations of Cost Shaping

In this section we argue that *both* the cost weights c and the risk measure $\rho(\cdot)$ are *necessary* to reasonably approximate diverse risk-sensitive behaviors. In Table 1, we provide the learned feature weights from RS-IRL and RN-IRL for features $\{\phi_1, \phi_2, \phi_3, \phi_4\}$ for the participants in case studies #1 and #2. These are the four features that dominate the along-track behavior of the participants.



(a) Expected (w.r.t. stochastic policy) prediction errors $\Delta v_{x,\mathrm{rel},t}$ from RS-IRL and RN-IRL for each 1.5 s trajectory segment.

(b) Percentage improvement in $\Delta v_{x,\mathrm{rel},t}$ for the RS-IRL model over RN-IRL for each 1.5 s trajectory segment.

Figure 23: Comparison of the $\Delta v_{x,\mathrm{rel},t}$ prediction errors (normalized by car length) for the RS-IRL and RN-IRL models for a risk-neutral participant. The two models perform on par with each other.

	Case Study #1		Case Study #2	
Feature weight	RS-IRL	RN-IRL	RS-IRL	RN-IRL
$c(1)$	0.0918	0.0466	0.0748	0.1884
$c(2)$	0.5174	0.5589	0.6354	0.2313
$c(3)$	0.0993	0.1052	0.1864	0.3170
$c(4)$	0.2352	0.2330	0.0562	0.0624

Table 1: Comparison of the inferred cost weights from RS-IRL and RN-IRL for the participants in case studies #1 and 2.

Notice that in spite of the extremely similar cost weights for the participant in case study #1, the improvement in performance using RS-IRL is substantial. For the participant in case study #2, a difference in the cost weights *and* the use of a risk measure were needed to obtain the observed performance boost with RS-IRL. This clearly establishes the benefits of risk sensitive inference, particularly highlighting the deficiency of only using cost shaping (in general, risk-neutral algorithms) due to its inability to cope with more nuanced manifestations of uncertainty in decision-making.

5.3.5 Summary

Table 2 presents the average (over the 51 1.5 s segments in the test trajectory) percentage improvement (RN-IRL to RS-IRL) in the prediction errors, i.e., Δx_{rel}, Δy_{rel}, $\Delta v_{x,\mathrm{rel}}$ and $\Delta v_{y,\mathrm{rel}}$. As expected, the RS-IRL predictions are almost always better than those provided by RN-IRL, with as much as 22.0% improvement in x_{rel} and 23.1% improvement in $v_{x,\mathrm{rel}}$. Regarding errors in y_{rel} and $v_{y,\mathrm{rel}}$, RS-IRL and RN-IRL perform comparably (absolute errors were in the range 0.19 – 0.42 m) since the primary source of risk-aversion stems from the leader's acceleration and deceleration rather than its lateral motion. In the cases with noticeable improvement, either in position or velocity (i.e., participants #2, 5, 6, 9), the better predictions were a result of the RS-IRL model more accurately representing participants with higher levels of risk- and/or ambiguity-aversion. Indeed the first two detailed case studies presented earlier correspond to participants #2 and #5. In contrast, for more risk-neutral participants (e.g., the last presented case study corresponding

to participant #4), the performance improvements were either less pronounced or the two models performed comparably.

Participant #	1	2	3	4	5	6	7	8	9	10
Δx_{rel} (T=2)	7.5	22.0	3.3	2.6	13.3	14.9	3.9	9.7	18.6	10.1
Δy_{rel} (T=2)	10.7	22.2	-2.7	12.0	4.8	9.5	8.4	7.3	16.3	21.4
$\Delta v_{x,\text{rel}}$ (T=2)	12.7	23.1	5.0	-0.3	14.3	13.0	8.1	10.5	17.1	10.4
$\Delta v_{y,\text{rel}}$ (T=2)	8.7	22.9	-3.0	13.7	0.2	6.0	9.1	3.0	14.9	22.5

Table 2: Average percentage improvement (over 51 segments in the test trajectory) in prediction errors for RS-IRL over RN-IRL. The RS-IRL predictions with $T = 2$ for x_{rel} and $v_{x,\text{rel}}$ are more accurate than those for RN-IRL for all but one participant, with as much as 23.1% average improvement. The most pronounced improvements (highlighted in red) corresponded to participants with higher levels of risk- and/or ambiguity-aversion, some of whom are studied in detail in the case studies. The improvements in the lateral direction are less significant since the primary source of risk and ambiguity was in the longitudinal dynamics.

6 Discussion and Conclusions

We have presented an approach for IRL that explicitly accounts for risk *and* ambiguity sensitivity in experts. We proposed a flexible modeling framework based on coherent risk measures that allows us to capture an entire spectrum of risk assessments from risk-neutral to worst-case for a rich class of static and dynamic decision-making settings. We developed efficient LP based non-parametric algorithms for static, and likelihood based semi-parametric algorithms for dynamic decision making settings. Notably, we significantly improved the modeling framework in (Majumdar et al., 2017) for dynamic decision making, and verified the performance improvements from RS-IRL despite transitioning from the exact LP iteration to semi-parametric likelihood based algorithms. The proposed inference framework was rigorously evaluated on a realistic simulated driving game with ten participants and shown to be able to infer and mimic qualitatively different driving styles ranging from risk-neutral to highly risk-averse in a data-efficient manner, while more accurately capturing participant behavior than with a risk-neutral model. Most importantly, by performing inference using the dual representation of coherent risk measures, we retain the generality to be able to recover any risk measure within such a class of risk measures, *without* assuming any a priori knowledge (e.g., fixed disutility function and/or risk measure).

Throughout this work, we assumed a *discrete* model of uncertainty for both the static and dynamic decision-making settings. While one would like to be able to address large or continuous sets of disturbances, we believe that a hierarchical representation of uncertainty is a more tractable approach. For instance, at the higher-level, one reasons about various uncertain *modes* of operation (e.g., the random erratic car maneuvers). At the lower-level, conditional on a given mode (e.g., deceleration), one may consider continuous models of uncertainty (e.g., the set of all robot deceleration profiles). The overall framework thus constitutes a mixture model. At the continuous lower-level of uncertainty (e.g., due to the natural variance in demonstrations), aspects such as risk-sensitivity are less relevant. Thus, this work studies risk-sensitivity at the *higher-level* hierarchy of decision making where discrete/modal models of uncertainty induce more nuanced behavior.

This paper opens several directions for future research. First, as in the majority of IRL literature, we hand-picked features for the driving game. While performance on the test trajectory given $\sim$ 1 minute of training data supported our choice of features, incorporating risk-sensitivity

in large-scale IRL algorithms requires automatic feature extraction. There has been some recent work on using deep neural nets within the MaxEnt IRL framework (Wulfmeier et al., 2015) as well as using general non-linear cost representations (Finn et al., 2016). A promising area of future research then is to embed the semi-parametric approach proposed in this paper within deep cost networks to yield RS-IRL algorithms for high-dimensional systems.

Second, our inference framework assumes that the human expert is subject to an independent (non-interactive) source of disturbance. The natural extension therefore is to modularize the entire risk-sensitive IRL algorithm within a game-theoretic *interactive* setting involving multiple human agents and robotic systems. The key challenge here is to efficiently balance offline and online learning (see e.g., (Sadigh et al., 2016b; Waugh et al., 2011)), to enable the autonomous robot to actively infer intent and risk-sensitive preferences for the human agents, and use the resulting information to consequently *influence* the human agents.

Finally, as a direct extension of the driving game and the game-theoretic adaptation of this work, we plan on testing our algorithms on an autonomous car testbed.

We believe that the approach described here along with the indicated future directions represent an important step towards endowing future robotic systems with the ability to predict, infer, and mimic risk-sensitive behavior, which is crucial for safety-critical applications where humans and robots interact.

Acknowledgments

The authors were partially supported by the Office of Naval Research, Science of Autonomy Program, under Contract N00014-15-1-2673, and by the Toyota Research Institute ("TRI"). This article solely reflects the opinions and conclusions of its authors and not ONR, TRI or any other Toyota entity.

The authors would also like to acknowledge the contributions of Ajay Mandlekar towards the conference version of this paper presented at RSS 2017.

References

Abbeel P and Ng AY (2004) Apprenticeship learning via inverse reinforcement learning. In: *Int. Conf. on Machine Learning*.

Abbeel P and Ng AY (2005) Exploration and apprenticeship learning in reinforcement learning. In: *Int. Conf. on Machine Learning*.

Acerbi C (2002) Spectral measures of risk: A coherent representation of subjective risk aversion. *Journal of Banking & Finance* 26(7): 1505–1518.

Acerbi C and Tasche D (2002) On the coherence of expected shortfall. *Journal of Banking & Finance* 26(7): 1487–1503.

Allais M (1953) Le comportement de l'homme rationnel devant le risque: critique des postulats et axiomes de lécole américaine. *Econometrica* 21(4): 503–546.

ApS M (2017) MOSEK optimization software. Available at https://mosek.com/.

Artzner P, Delbaen F, Eber JM and Heath D (1999) Coherent measures of risk. *Mathematical Finance* 9(3): 203–228.

Axelrod A, Carlone L, Chowdhary G and Karaman S (2016) Data-driven prediction of EVAR with confidence in time-varying datasets. In: *IEEE Conf. on Decision and Control*.

Barberis NC (2013) Thirty years of prospect theory in economics: A review and assessment. *Journal of Economic Perspectives* 27(1): 173–195.

Bäuerle N and Ott J (2011) Markov decision processes with average-value-at-risk criteria. *Mathematics of Operations Research* 74(3): 361–379.

Burke JV, Lewis AS and Overton ML (2005) A robust gradient sampling algorithm for nonsmooth nonconvex optimization. *SIAM Journal on Optimization* 15(3): 751–779.

Carton D, Nitsch V, Meinzer D and Wollherr D (2016) Towards assessing the human trajectory planning horizon. *PLoS ONE* 11(12): e0167021.

Chen X (2012) Smoothing methods for nonsmooth, nonconvex minimization. *Mathematical Programming* 134(1): 71–99.

Chow Y and Pavone M (2014) A framework for time-consistent, risk-averse model predictive control: Theory and algorithms. In: *American Control Conference*.

Chow Y, Tamar A, Mannor S and Pavone M (2015) Risk-sensitive and robust decision-making: a CVaR optimization approach. In: *Advances in Neural Information Processing Systems*.

Eichhorn A and Römisch W (2005) Polyhedral risk measures in stochastic programming. *SIAM Journal on Optimization* 16(1): 69–95.

Ellsberg D (1961) Risk, ambiguity, and the savage axioms. *The Quarterly Journal of Economics* 75(4): 643–669.

Englert P and Toussaint M (2015) Inverse KKT learning cost functions of manipulation tasks from demonstrations. In: *Int. Symp. on Robotics Research*.

Filar JA, Kallenberg LCM and Lee HM (1989) Variance-penalized Markov decision processes. *Mathematics of Operations Research* 14(1): 147–161.

Finn C, Levine S and Abbeel P (2016) Guided cost learning: Deep inverse optimal control via policy optimization. In: *Int. Conf. on Machine Learning*.

Geibel P and Wysotzki F (2005) Risk-sensitive reinforcement learning applied to control under constraints. *Journal of Artificial Intelligence Research* 24(1): 81–108.

Gilboa I and Marinacci M (2016) Ambiguity and the Bayesian paradigm. In: *Readings in Formal Epistemology*, first edition, chapter 21.

Gilboa I and Schmeidler D (1989) Maxmin expected utility with non-unique prior. *Journal of Mathematical Economics* 18(2): 141–153.

Glimcher P and Fehr E (2014) *Neuroeconomics*. Second edition. Elsevier.

Gul F and Pesendorfer W (2014) Expected uncertain utility theory. *Econometrica* 82(1): 1–39.

Hey JD, Lotito G and Maffioletti A (2010) The descriptive and predictive adequacy of theories of decision making under uncertainty/ambiguity. *Journal of Risk and Uncertainty* 41(2): 81–111.

Howard R and Matheson J (1972) Risk-sensitive Markov decision processes. *Management Science* 8(7): 356–369.

Hsu M, Bhatt M, Adolphs R, Tranel D and Camerer CF (2005) Neural systems responding to degrees of uncertainty in human decision-making. *Science* 310(5754): 1680–1683.

Kahneman D and Tversky A (1979) Prospect theory: An analysis of decision under risk. *Econometrica* : 263–291.

Kolter JZ, Abbeel P and Ng AY (2007) Hierarchical apprenticeship learning with application to quadruped locomotion. In: *Advances in Neural Information Processing Systems*.

Kretzschmar H, Spies M, Sprunk C and Burgard W (2016) Socially compliant mobile robot navigation via inverse reinforcement learning. *Int. Journal of Robotics Research* 35(11): 1289–1307.

Kuderer M, Gulati S and Burgard W (2015) Learning driving styles for autonomous vehicles from demonstration. In: *Proc. IEEE Conf. on Robotics and Automation*.

Lanza A, Morigi S, Selesnick I and Sgallari F (2017) Nonconvex nonsmooth optimization via convexnonconvex majorizationminimization. *Numerische Mathematik* 136(2): 343–381.

Levine S and Koltun V (2012) Continuous inverse optimal control with locally optimal examples. In: *Int. Conf. on Machine Learning*.

Löfberg J (2004) YALMIP : A toolbox for modeling and optimization in MATLAB. In: *IEEE Int. Symp. on Computer Aided Control Systems Design*.

Majumdar A and Pavone M (2017) How should a robot assess risk? Towards an axiomatic theory of risk in robotics. In: *Int. Symp. on Robotics Research*.

Majumdar A, Singh S, Mandlekar A and Pavone M (2017) Risk-sensitive inverse reinforcement learning via coherent risk models. In: *Robotics: Science and Systems*.

Mihatsch O and Neuneier R (2002) Risk-sensitive reinforcement learning. *Machine Learning* 49(2): 267–290.

Mombaur K, Truong A and Laumond JP (2010) From human to humanoid locomotion—an inverse optimal control approach. *Autonomous Robots* 28(3): 369–383.

Ng A and Russell S (2000) Algorithms for inverse reinforcement learning. In: *Int. Conf. on Machine Learning*.

Nilim A and El Ghaoui L (2005) Robust control of Markov decision processes with uncertain transition matrices. *Operations Research* 53(5): 780–798.

Osogami T (2012) Robustness and risk-sensitivity in Markov decision processes. In: *Advances in Neural Information Processing Systems*.

Park T and Levine S (2013) Inverse optimal control for humanoid locomotion. In: *Robotics: Science and Systems Workshop on Inverse Optimal Control and Robotic Learning from Demonstration*.

Petrik M and Subramanian D (2012) An approximate solution method for large risk-averse Markov decision processes. In: *Proc. Conf. on Uncertainty in Artificial Intelligence*.

Prashanth LA, Jie C, Fu M, Marcus S and Szepesvári C (2016) Cumulative prospect theory meets reinforcement learning: Prediction and control. In: *Int. Conf. on Machine Learning*.

Quiggin J (1982) A theory of anticipated utility. *Journal of Economic Behavior & Organization* 3(4): 323–343.

Rabin M (2000) Risk aversion and expected-utility theory: A calibration theorem. *Econometrica* 68(5): 1281–1292.

Ramachandran D and Amir E (2007) Bayesian inverse reinforcement learning. In: *International Joint Conference on Artificial Intelligence*.

Ratliff LJ and Mazumdar E (2017) Risk-sensitive inverse reinforcement learning via gradient methods. Available at: https://arxiv.org/abs/1703.09842.

Rockafellar RT (2007) Coherent approaches to risk in optimization under uncertainty. In: *OR Tools and Applications: Glimpses of Future Technologies*, chapter 3. INFORMS.

Rockafellar RT and Uryasev S (2000) Optimization of conditional value-at-risk. *Journal of Risk* 2: 21–41.

Rockafellar RT and Uryasev S (2002) Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance* 26(7): 1443–1471.

Russell S (1998) Learning agents for uncertain environments. In: *Proc. Computational Learning Theory*.

Ruszczyński A (2010) Risk-averse dynamic programming for Markov decision process. *Mathematical Programming* 125(2): 235–261.

Sadigh D, Sastry S, Seshia SA and Dragan SD (2016a) Planning for autonomous cars that leverage effects on human actions. In: *Robotics: Science and Systems*.

Sadigh D, Sastry SS, Seshia SA and Dragan A (2016b) Information gathering actions over human internal state. In: *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*.

Shapiro A (2009) On a time consistency concept in risk averse multi-stage stochastic programming. *Operations Research Letters* 37(3): 143–147.

Shapiro A, Dentcheva D and Ruszczyński A (2014) *Lectures on stochastic programming: Modeling and theory*. Second edition. SIAM.

Shen Y, Tobia MJ, Sommer T and Obermayer K (2014) Risk-sensitive reinforcement learning. *Neural Computation* 26(7): 1298–1328.

Tamar A, Chow Y, Ghavamzadeh M and Mannor S (2016) Sequential decision making with coherent risk. *IEEE Transactions on Automatic Control* 62(7): 3323–3338.

Tamar A, Di Castro D and Mannor S (2012) Policy gradients with variance related risk criteria. In: *Int. Conf. on Machine Learning*.

VIRES Simulationstechnologie GmbH (2017) VTD - Virtual Test Drive. Available at https://vires.com/vtd-vires-virtual-test-drive/.

von Neumann J and Morgenstern O (1944) *Theory of Games and Economic Behavior*. Princeton University Press.

Waugh K, Ziebart BD and Bagnell JA (2011) Computational rationalization: The inverse equilibrium problem. In: *Int. Conf. on Machine Learning*.

Wu C and Yuanlie L (1999) Minimizing risk models in markov decision process with policies depending on target values. *Journal of Mathematical Analysis and Applications* 231(1): 47–67.

Wulfmeier M, Ondruska P and Posner I (2015) Maximum entropy deep inverse reinforcement learning. Available at: https://arxiv.org/abs/1507.04888.

Xu H and Mannor S (2010) Distributionally robust Markov decision processes. In: *Advances in Neural Information Processing Systems*.

Yaari ME (1987) The dual theory of choice under risk. *Econometrica* 55(1): 95–115.

Ziebart BD, Maas A, Bagnell JA and Dey AK (2008) Maximum entropy inverse reinforcement learning. In: *Proc. AAAI Conf. on Artificial Intelligence*.

Ziebart BD, Ratliff N, Gallagher G, Mertz C, Peterson K, Bagnell JA, Hebert M, Key AK and Srinivasa S (2009) Planning-based prediction for pedestrians. In: *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*.

Zucker M, Bagnell JA, Atkeson CG and Kuffner J (2010) An optimization approach to rough terrain locomotion. In: *Proc. IEEE Conf. on Robotics and Automation*.

Appendix A Theoretical Guarantees

A.1 Proof of Theorem 3

In order to prove the algorithm's consistency, we need to establish a few intermediate results. Since $\mathcal{P}_\infty$ is an intersection of convex (respectively, compact) sets, it is also convex (respectively, compact). Moreover, since each $\mathcal{P}_D$ contains the expert's polytope $\mathcal{P}$, then $\mathcal{P}_\infty$ is also an outer approximation of $\mathcal{P}$. In particular, $\mathcal{P}_\infty$ is not empty.

Denote with d_H the Hausdorff distance between subsets of $\mathbb{R}^L$ associated with the Euclidean norm $||.||$, i.e., for two subsets A and B of $\mathbb{R}^L$,

$$d_H(A,B) := \max\left\{\sup_{b\in B}\inf_{a\in A}\|a-b\|,\ \sup_{a\in A}\inf_{b\in B}\|a-b\|\right\}. \tag{37}$$

The Hausdorff distance defines a metric on the set of non-empty compact subsets of $\mathbb{R}^L$, that we use to measure the distance between risk envelopes. The sequence $\mathcal{P}_d$ is non-increasing (for set inclusion). Therefore, $\{d_H(\mathcal{P}_d, \mathcal{P}_\infty)\}_{d\geq 1}$ is a non-increasing sequence of non-negative real numbers. In particular, we have the following result:

Lemma 4 (Convergence in Hausdorff metric)**.** *The sequence $\{d_H(\mathcal{P}_d, \mathcal{P}_\infty)\}$ goes to 0 when $d \to \infty$.*

Lemma 5 (Compact uniform convergence)**.** *Consider a sequence $\{\mathcal{Q}_n\}_{n\geq 1}$ of compact convex subsets of Δ^L such that for all $n \geq 1$, $\mathcal{P}_\infty \subseteq \mathcal{Q}_n$ and $\lim_{n\to\infty} d_H(\mathcal{Q}_n, \mathcal{P}_\infty) = 0$. Consider also a sequence of states $\{x_n\}_{n\geq 1}$ such that $x_n \to x^*$ when $n \to \infty$. Define the functions $\varphi_n(u) := \max_{v\in\mathcal{Q}_n} v^T g(x_n, u)$, $\varphi_{n,\infty}(u) := \max_{v\in\mathcal{P}_\infty} v^T g(x_n, u)$ and $\varphi(u) := \max_{v\in\mathcal{P}_\infty} v^T g(x^*, u)$. Then, for any compact set $\mathcal{K} \subseteq \mathcal{U}$:*

$$\lim_{n\to\infty} \sup_{u\in\mathcal{K}} |\varphi_n(u) - \varphi(u)| = 0. \tag{38}$$

Proof. Fix $u \in \mathcal{U}$. For any $n \geq 1$, denote with $v_n \in \mathcal{Q}_n$ a point such that $\varphi_n(u) = v_n^T g(x_n, u)$, and with $v_{n,\infty} \in \mathcal{P}_\infty$ a point such that $\varphi_{n,\infty}(u) = v_{n,\infty}^T g(x_n, u)$. Let Γ_∞ be the projection operator onto the compact convex set $\mathcal{P}_\infty$. Then,

$$\Gamma_\infty(v_n)^T g(x_n, u) \leq v_{n,\infty}^T g(x_n, u) \leq v_n^T g(x_n, u). \tag{39}$$

The second inequality in (39) results from the fact that $\mathcal{P}_\infty \subseteq \mathcal{Q}_n$. By Cauchy-Schwarz inequality,

$$|\left(\Gamma_\infty(v_n) - v_n\right)^T g(x_n, u)| \leq ||\Gamma_\infty(v_n) - v_n||_2 ||g(x_n, u)||_2. \tag{40}$$

Since $||\Gamma_\infty(v_n) - v_n||_2 \leq d_H(\mathcal{Q}_n, \mathcal{P}_\infty)$ and $\lim_{n\to\infty} x_n = x^*$, we get that the left-hand side (LHS) in (40) tends to 0 when $n \to \infty$, which implies that

$$\lim_{n\to\infty} \varphi_n(u) - \varphi_{n,\infty}(u) = 0. \tag{41}$$

Since g is continuous with respect to the state variable x and $\mathcal{P}_\infty$ is compact, we get that $x \mapsto \max_{v\in\mathcal{P}_\infty} v^T g(x, u)$ is also continuous with respect to x. Therefore, $\lim_{n\to\infty} \varphi_{n,\infty}(u) = \varphi(u)$. Using (41), $\lim_{n\to\infty} \varphi_n(u) = \varphi(u)$. But, by Theorem 10.8 in (Rockafellar, 2007), pointwise convergence of a sequence of convex functions over $\mathcal{U}$ implies uniform convergence over any compact set $\mathcal{K} \subseteq \mathcal{U}$, which is the desired result. □

Since we assumed the cost functions to be strictly convex with respect to u, the risk-sensitive optimization problem admits a unique optimal control.

Lemma 6 (Strict convexity of risk)**.** *For any compact subset $\mathcal{B} \subseteq \Delta^L$ and at any state x, the function $u \mapsto \max_{v\in\mathcal{B}} v^T g(x, u)$ is strictly convex. In particular, it admits a unique minimizer.*

Proof. Fix a state x. Take $u_1, u_2 \in \mathcal{U}$ and $\alpha \in [0, 1]$. Denote $u_\alpha = (1-\alpha)u_1 + \alpha u_2$. Since $\mathcal{B}$ is compact, there exists $\bar{v} \in \mathcal{B}$ such that: $\bar{v}^T g(x, u_\alpha) = \max_{v\in\mathcal{B}} v^T g(x, u_\alpha)$. By strict convexity of $u \mapsto \bar{v}^T g(x, u)$, it follows that: $\bar{v}^T g(x, u_\alpha) < (1-\alpha)\bar{v}^T g(x, u_1) + \alpha \bar{v}^T g(x, u_2)$. Taking the worst-case for both terms of the previous inequality's right-hand side, we get that: $\bar{v}^T g(x, u_\alpha) < (1-\alpha)\max_{v\in\mathcal{B}} v^T g(x, u_1) + \alpha \max_{v\in\mathcal{B}} v^T g(x, u_2)$, which proves strict convexity of the function $u \mapsto \max_{v\in\mathcal{B}} v^T g(x, u)$. Since the latter function has, by assumption, bounded level sets, it admits a minimizer, which is unique by strict convexity. □

Lemma 7 (Optimal control convergence)**.** *Define the sequence of functions φ_n and φ as in Lemma 5. Denote $u_n := \operatorname{argmin}_{u\in\mathcal{U}} \varphi_n(u)$ and $u^* := \operatorname{argmin}_{u\in\mathcal{U}} \varphi(u)$ (each of these minima are unique by strict convexity). Then:*

$$\lim_{n\to\infty} u_n = u^*. \tag{42}$$

Proof. Let $\tau_n := \min_{u\in\mathcal{U}} \varphi_n(u)$ and $\tau := \min_{u\in\mathcal{U}} \varphi(u)$. By construction, the control set $\mathcal{U}$ is a compact convex set. Thus, by Lemma 5, the function φ_n converges to φ uniformly over $\mathcal{U}$. Thus, for any given $\epsilon > 0$, there exists an $n_0(\epsilon) \in \mathbb{N}$ such that for all $n > n_0(\epsilon)$,

$$|\varphi_n(u) - \varphi(u)| \leq \epsilon \quad \forall u \in \mathcal{U}.$$

It follows that for $n > n_0(\epsilon)$, we have:

$$\begin{aligned} |\varphi_n(u^*) - \varphi(u^*)| &= |\varphi_n(u^*) - \tau| \leq \epsilon \\ |\varphi_n(u_n) - \varphi(u_n)| &= |\tau_n - \varphi(u_n)| \leq \epsilon. \end{aligned} \tag{43}$$

Furthermore,

$$\tau_n - \varphi(u_n) \leq \tau_n - \tau \leq \varphi_n(u^*) - \tau,$$

since $\varphi(u) \geq \tau$ for all $u \in \mathcal{U}$ and $\tau_n \leq \varphi_n(u)$ for all $u \in \mathcal{U}$. Combining this with eq. (43), we have:

$$-\epsilon \leq \tau_n - \tau \leq \epsilon, \quad \forall n > n_0(\epsilon).$$

Thus, $\tau_n \to \tau$ as $n \to \infty$. We proceed by contradiction to prove (42). Assume that $\{u_n\}_{n\geq 1}$ does not converge to u^*. Without loss of generality, assume that there exists some $\eta > 0$ such that for all n, $||u_n - u^*|| \geq \eta$. Define u'_n and $\alpha_n \in [0, 1]$ such that $u'_n = \alpha_n u_n + (1 - \alpha_n)u^*$, and $||u^* - u'_n|| = \frac{\eta}{2}$. By convexity of φ_n:

$$\varphi_n(u'_n) \leq \alpha_n \varphi_n(u_n) + (1 - \alpha_n)\varphi_n(u^*) = \alpha_n \tau_n + (1 - \alpha_n)\varphi_n(u^*). \tag{44}$$

By the pointwise convergence property of φ_n, we have that $\lim_{n\to\infty} \varphi_n(u^*) = \varphi(u^*) = \tau$. Furthermore, we have also established that $\lim_{n\to\infty} \tau_n = \tau^*$. Thus, the RHS in eq. (44) converges to $\varphi(u^*)$. For the LHS, assume that the sequence $\{u'_n\}_{n\geq 1}$ converges to u' (or consider a converging subsequence, which is allowed since $\mathcal{U}$ is compact). Then, by continuity of φ and uniform convergence of φ_n, it follows that: $\lim_{n\to\infty} \varphi_n(u'_n) = \varphi(u')$. Using (44), we get: $\varphi(u') \leq \varphi(u^*)$. But $||u' - u^*|| = \frac{\eta}{2}$ and by uniqueness of the minimum, this is a contradiction. □

We are now ready to prove Theorem 3.

Proof. (Theorem 3). Fix any $x^* \in \mathcal{S}$ and choose a subsequence of $\{\mathcal{P}_d\}_{d\geq 1}$ (that we still denote $\{\mathcal{P}_d\}$ for simplicity) such that $x^{*,d} \to x^*$. For any $d \geq 1$ and corresponding demonstration $(x^{*,d}, u^{*,d})$, according to the update of the outer approximation $\mathcal{P}_d$ in Algorithm 1,

$$u(\mathcal{P}_d, x^{*,d}) = u(\mathcal{P}, x^{*,d}). \tag{45}$$

We justify (45). Given the demonstration pair $(x^{*,d}, u^{*,d})$ where by definition, $u^{*,d} = u(\mathcal{P}, x^{*,d})$, let $\bar{v} \in \mathcal{P}_{d-1}$, $\bar{\sigma}_+$, $\bar{\sigma}_-$ be solutions of:

$$\begin{aligned} \max_{\substack{v \in \mathcal{P}_{d-1} \\ \sigma_+, \sigma_- \geq 0}} \quad & g(x^{*,d}, u^{*,d})^T v \\ s.t. \quad & 0 = \nabla_{u(j)} g(x,u)^T v \big|_{x^{*,d}, u^{*,d}} + \sigma_+(j), \forall j \in \mathcal{J}^+ \\ & 0 = \nabla_{u(j)} g(x,u)^T v \big|_{x^{*,d}, u^{*,d}} - \sigma_-(j), \forall j \in \mathcal{J}^- \\ & 0 = \nabla_{u(j)} g(x,u)^T v \big|_{x^{*,d}, u^{*,d}}, \forall j \notin \mathcal{J}^+, j \notin \mathcal{J}^- \\ & \sigma_+(j) = 0, \ \sigma_-(j) = 0, \quad \forall j \notin \mathcal{J}^+, j \notin \mathcal{J}^- \end{aligned} \tag{46}$$

where $\mathcal{J}^+ = \{j \in \{1, \dots, m\} \,|\, u^{*,d}(j) = u^+(j)\}$ and $\mathcal{J}^- = \{j \in \{1, \dots, m\} \,|\, u^{*,d}(j) = u^-(j)\}$. According to Algorithm 1, $\mathcal{P}_d = \{v \in \mathcal{P}_{d-1} \,|\, v^T g(x^{*,d}, u^{*,d}) \leq \bar{v}^T g(x^{*,d}, u^{*,d})\}$. Moreover, $\bar{v} \in \mathcal{P}_d$, which implies:

$$\max_{v \in \mathcal{P}_d} v^T g(x^{*,d}, u^{*,d}) = \bar{v}^T g(x^{*,d}, u^{*,d}). \tag{47}$$

The set of equations given by the constraints of Problem (46), with v, σ_+ and σ_- fixed and respectively equal to $\bar{v}$, $\bar{\sigma}_+$ and $\bar{\sigma}_-$, are exactly the optimality conditions for the solution of the following convex optimization problem:

$$\min_{u \in \mathcal{U}} \bar{v}^T g(x^{*,d}, u). \tag{48}$$

Since $u^{*,d}$ satisfies those optimality conditions and using (47), we have:

$$\begin{aligned} \max_{v \in \mathcal{P}_d} v^T g(x^{*,d}, u^{*,d}) &= \bar{v}^T g(x^{*,d}, u^{*,d}) \\ &= \min_{u \in \mathcal{U}} \bar{v}^T g(x^{*,d}, u) \\ &\leq \min_{u \in \mathcal{U}} \max_{v \in \mathcal{P}_d} v^T g(x^{*,d}, u) \\ &\leq \max_{v \in \mathcal{P}_d} v^T g(x^{*,d}, u^{*,d}) \end{aligned} \tag{49}$$

Hence, all inequalities in (49) are equalities. In particular,

$$\min_{u \in \mathcal{U}} \max_{v \in \mathcal{P}_d} v^T g(x^{*,d}, u) = \max_{v \in \mathcal{P}_d} v^T g(x^{*,d}, u^{*,d}) \tag{50}$$

. By uniqueness of the minimum as given by Lemma 6, it follows that $u^{*,d} = u(\mathcal{P}_d, x^{*,d})$. From Lemma 7, we have that $\lim_{d \to \infty} u\left(\mathcal{P}_d, x^{*,d}\right) = u\left(\mathcal{P}_\infty, x^*\right)$. From equation (45), we get that $\lim_{d \to \infty} u\left(\mathcal{P}_d, x^{*,d}\right) = u\left(\mathcal{P}, x^*\right)$. Combining the two previous observations, we get the desired result, i.e., $u\left(\mathcal{P}, x^*\right) = u\left(\mathcal{P}_\infty, x^*\right)$. □

Appendix B Derivation of Prepare-React Policy Likelihood

Recall the multi-stage optimization problem objective, repeated here for convenience:

$$C_{0:N-n_d} + \rho_0 \Big(C_{N-n_d+1:N-1} + C_{N:2N-n_d} + \rho_1 \left(C_{2N-n_d+1:2N-1} + \dots + \rho_{T-1} \left(C_{TN-n_d+1:TN-1} \right) \cdots \right) \Big).$$

For a "prepare" – "react" policy $\hat{\pi}_t$ at stage t, let $\hat{\pi}_{t|\mathfrak{p}}$ denote the "prepare" portion and $\hat{\pi}_{t|\mathfrak{r}}$ denote the "react" portion. Then, we can re-write the objective above to show explicit dependence as follows:

$$C_{0:N-n_d}(\cdot, \hat{\pi}_{0|\mathfrak{p}}) + \rho_0\Bigg(C_{N-n_d+1:N-1}(\cdot, \hat{\pi}_{0|\mathfrak{r}}) + C_{N:2N-n_d}(\cdot, \hat{\pi}_{1|\mathfrak{p}}) + \\ \rho_1\big(C_{2N-n_d+1:2N-1}(\cdot, \hat{\pi}_{1|\mathfrak{r}}) + \cdots + \rho_{T-1}\left(C_{TN-n_d+1:TN-1}(\cdot, \hat{\pi}_{T-1|\mathfrak{r}})\right)\cdots\big)\Bigg).$$

Note that the expression within the large brackets is a random variable in $\mathbb{R}^L$, indexed by all possible realizations of w_0', and a function of the first "prepare" sequence $\hat{\pi}_{0|\mathfrak{p}}$. Thus, for a given "prepare" sequence $\hat{\pi}_{0|\mathfrak{p}}$ and first disturbance mode w_0', define the optimal tail cost as a function of $\hat{\pi}_{0|\mathfrak{r}}$:

$$\tau[\hat{\pi}_{0|\mathfrak{p}}, w_0'](\hat{\pi}_{0|\mathfrak{r}}) := C_{N-n_d+1:N-1}(\cdot, \hat{\pi}_{0|\mathfrak{r}}) + \\ \min_{\substack{\hat{\pi}_t \\ t\in[1,T-1]}} \rho_1\big(C_{N:2N-1}(\cdot, \hat{\pi}_1) + \cdots + \rho_{T-1}(C_{(T-1)N:TN-1}(\cdot, \hat{\pi}_{T-1}))\big).$$

Then, we define the *conditional* distribution for the "react" sequence corresponding to disturbance mode w_0', given the first "prepare" sequence, as

$$\Pr(\hat{\pi}_{0|\mathfrak{r}} \mid \hat{\pi}_{0|\mathfrak{p}}\ ; w_0') \propto \exp\left(-\tau[\hat{\pi}_{0|\mathfrak{p}}, w_0'](\hat{\pi}_{0|\mathfrak{r}})\right).$$

The distribution for the first "prepare" sequence is then given by

$$\Pr(\hat{\pi}_{0|\mathfrak{p}}) \propto \exp\left(-\left[C_{0:N-n_d}(\cdot, \hat{\pi}_{0|\mathfrak{p}}) + \rho^r\left(\operatorname*{softmin}_{\hat{\pi}_{0|\mathfrak{r}}} \tau[\hat{\pi}_{0|\mathfrak{p}}, w_0'](\hat{\pi}_{0|\mathfrak{r}})\right)\right]\right),$$

where we use softmin in place of min to ensure differentiability. Thus, for an observed "prepare" – "react" sequence $\hat{\pi}_0^*$ associated with the observed disturbance mode w_0^*, we obtain

$$\Pr(\hat{\pi}_0^*) = \Pr(\hat{\pi}_{0|\mathfrak{p}}^*) \cdot \Pr(\hat{\pi}_{0|\mathfrak{r}}^* \mid \hat{\pi}_{0|\mathfrak{p}}^*\ ; w_0^*).$$

Appendix C Likelihood Gradient Computations

Define

$$\sigma[w^*_{-1|tN}](\hat{u}) := \frac{\exp\left(-\beta\tilde{\tau}[w^*_{-1|tN}](\hat{u})\right)}{\sum_{\hat{u}'}\exp\left(-\beta\tilde{\tau}[w^*_{-1|tN}](\hat{u}')\right)} \tag{51}$$

to be the probability of choosing action trajectory $\hat{u}$ at time-step tN as assumed by the Boltzmann likelihood model in (29). Then, the gradient of the log-likelihood in (31) with respect to parameter $s \in \{r, c\}$ is given by

$$\frac{\beta}{|\mathcal{T}^*|}\sum_{\hat{u}_t^*\in\mathcal{T}^*}\left[\sum_{\hat{u}\neq\hat{u}_t^*}\sigma[w^*_{-1|tN}](\hat{u})\nabla_s\tilde{\tau}[w^*_{-1|tN}](\hat{u}) + \left(\sigma[w^*_{-1|tN}](\hat{u}_t^*) - 1\right)\nabla_s\tilde{\tau}[w^*_{-1|tN}](\hat{u}_t^*)\right]. \tag{52}$$

For notational clarity, we use t to denote the t^{th} N-step segment in the demonstrated trajectory $\mathcal{T}^*$ and t' as the stage-wise index within the multi-step planning problem. From equations (26), (27), and (28), we see that the derivative of $\tilde{\tau}[w^*_{-1|tN}](\hat{u})$ can be computed through a recursive implementation of the chain rule, starting from the terminal stage. In the event that all nested LPs are non-degenerate, we obtain the following recursive set of equations for computing the gradients.

Terminal Stage: From eq. (26), $\tilde{\tau}[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}](\hat{u})$ is the optimal value of the LP:

$$\max_{v \in \mathcal{P}_r} \left(g(w'_T, \hat{u}; c)\right)^T v, \tag{53}$$

where $g \in \mathbb{R}^L$ is the accumulated cost vector over the terminal stage, $C_{(T-1)N:TN-1}$, indexed by the terminal disturbance mode w'_T. Let v^* denote the optimal primal solution and λ^* the optimal dual variables for the constraints defined in (23). Then,

$$\nabla_r \tilde{\tau}[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}](\hat{u}) = -\lambda^*, \tag{54}$$

$$\nabla_c \tilde{\tau}[\boldsymbol{u}_{T-2}, \boldsymbol{\omega}_{T-2}](\hat{u}) = \sum_{j=1}^{L} v^*(j) \left[\Phi^{[j]}(\hat{u})\right], \tag{55}$$

where $\Phi^{[j]}(\hat{u})$ is the feature vector sum over N time steps corresponding to $C_{(T-1)N:TN-1}$, given[3] action trajectory $\hat{u}$ and disturbance $w^{[j]}$.

Recursion: For $t' \in \{0, \ldots, T-2\}$, $\tilde{\tau}[\boldsymbol{u}_{t'-1}, \boldsymbol{\omega}_{t'-1}](\hat{u})$ is the optimal value of the LP[4]:

$$\max_{v \in \mathcal{P}_r} \left(g(w'_t, \hat{u}; c) + \tilde{g}(w'_t, \hat{u}; c, r)\right)^T v,$$

where $g \in \mathbb{R}^L$ is the accumulated cost vector $C_{t'N:(t'+1)N-1}$, and $\tilde{g} \in \mathbb{R}^L$ is the vector of $\text{softmin}_{\hat{u}'}$ over the tail risk-sensitive costs, i.e., $\tilde{\tau}[\{\boldsymbol{u}_{t'-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t'-1}, w'_t\}](\hat{u}')$, indexed by the next disturbance mode w'_t, and parameterized with respect to r, c. Recall that $w'_{-1} = w^*_{-1|tN}$. Let v^* denote the optimal primal solution and λ^* the optimal dual variables for the constraints in eq. (23). Then,

$$\nabla_r \tilde{\tau}[\boldsymbol{u}_{t'-1}, \boldsymbol{\omega}_{t'-1}](\hat{u}) = \sum_{j=1}^{L} v^*(j) \, \nabla_r \tilde{g}(w^{[j]}, \hat{u}; c, r) - \lambda^*, \tag{56}$$

$$\nabla_c \, \tilde{\tau}[\boldsymbol{u}_{t'-1}, \boldsymbol{\omega}_{t'-1}](\hat{u}) = \sum_{j=1}^{L} v^*(j) \left[\Phi^{[j]}(\hat{u}) + \nabla_c \, \tilde{g}(w^{[j]}, \hat{u}; c, r)\right], \tag{57}$$

where $\Phi^{[j]}$ is the feature vector sum corresponding to $C_{t'N:(t'+1)N-1}$. The gradients of the softmin vector are given by:

$$\nabla_r \tilde{g}(w^{[j]}, \hat{u}; c, r) = \mathbb{E}_{\hat{u}' \sim \sigma\left[\tilde{\tau}[\{\boldsymbol{u}_{t'-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t'-1}, w^{[j]}\}]\right]} \left[\nabla_r \tilde{\tau}[\{\boldsymbol{u}_{t'-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t'-1}, w^{[j]}\}](\hat{u}')\right], \tag{58}$$

$$\nabla_c \tilde{g}(w^{[j]}, \hat{u}; c, r) = \mathbb{E}_{\hat{u}' \sim \sigma\left[\tilde{\tau}[\{\boldsymbol{u}_{t'-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t'-1}, w^{[j]}\}]\right]} \left[\nabla_c \tilde{\tau}[\{\boldsymbol{u}_{t'-1}, \hat{u}\}, \{\boldsymbol{\omega}_{t'-1}, w^{[j]}\}](\hat{u}')\right], \tag{59}$$

[3] For notational convenience, we omit the obvious dependence on state.

[4] For notational simplicity, take $\boldsymbol{u}_{-1} = \{\}$.

where $\sigma\left[\tilde{\tau}[\boldsymbol{u}_{t'}, \boldsymbol{\omega}_{t'}]\right]$ is the discrete (cost-based) Boltzmann distribution, i.e.,

$$\sigma\left[\tilde{\tau}[\boldsymbol{u}_{t'}, \boldsymbol{\omega}_{t'}]\right](\hat{u}') \propto \exp\left(-\tilde{\tau}[\boldsymbol{u}_{t'}, \boldsymbol{\omega}_{t'}](\hat{u}')\right).$$

For our experiments, we found a different form of the softmin function to be more numerically stable. In particular, we used

$$\underset{\beta}{\text{softmin}}\, f(x) = \mathbb{E}_{x \sim \sigma_\beta[f]} f(x),$$

where $\sigma_\beta[f]$ is the Boltzmann distribution defined with inverse temperature β as:

$$\sigma_\beta[f](x) \propto \exp\left(-\beta f(x)\right).$$

The gradients take the exact same form as (58) and (59) with σ replaced by σ_β.

Thus, to compute the gradient in (52) for the t^{th} demonstration, one would start with the terminal stage derivatives in (54) and (55) for the planning problem defined at time step tN, and proceed backwards inductively using (56)–(59) to arrive at $\nabla\tilde{\tau}[w^*_{-1|tN}](\hat{u})$.

Note that the derivation above assumes non-degeneracy of the LPs. It is readily observed that by the piecewise linearity of LPs with respect to the objective coefficients and constraint right-hand-sides, and the Lipschitz property of the softmin function, $\tilde{\tau}$ is locally Lipschitz. Consequently, the log-likelihood too is locally Lipschitz. Thus, by the Rademacher theorem, the log-likelihood is non-differentiable only over a Lebesgue set of measure zero. If, however, during the updates, any (nested) LP is primal degenerate (multiple dual optimal solutions) or dual degenerate (multiple primal optimal solutions), the log-likelihood is non-differentiable (despite directional-derivatives existing in all directions). Thus, in its full generality, the max likelihood problem corresponds to a non-convex, non-smooth optimization and at points of non-differentiability, one must do extra work to compute a suitable descent direction. Some proposed approaches in the literature include penalized smoothing (Chen, 2012), sampling-based estimation (Burke et al., 2005) to approximate the Clarke generalized subdifferential and compute a descent direction, and majorization-minimization (Lanza et al., 2017) to iteratively optimize an upper-bound on a minimization problem. Arguably, the field is an active area of research.

In our implementation, we implemented the following two heuristics to avoid non-degeneracy (and consequent non-differentiability of the log-likelihood):

- During the projection step of the projected gradient method for r, if a constraint $a_j^T v \leq b(j) - r(j)$ is found to be redundant, the parameter $r(j)$ is re-adjusted as $r(j) \leftarrow r(j) + 0.01$, provided that the resulting polytope is not empty. This has the effect of eliminating the possibility of redundant constraints at primal optimal vertices (thereby eliminating primal degeneracy).

- In the event that the objective vector is parallel to one of the bounding hyperplanes of the region $\mathcal{P}_r$ (i.e., dual degeneracy), we added a small distortion to the objective vector.